Sped. in Abb. Postale Gr.III/70%

Anno XVII - Numero 7 - Settembre 1988 - **L. 7.500**

CAMPIONATO
Diciotto squadre partita per partita

TASSA PAGATA PER CAMPIONE ALLEGATO

Gruppo Editoriale JCE

Radio elettronica & Computer

N. 7 Luglio/Agosto 1988

Rubriche:

Sommario

Caricate così i programmi della cassetta allegata:

*Riavvolgete il nastro e digitate **LOAD** seguito da **RETURN** sulla tastiera del C64 e **PLAY** sul registratore. Verrà caricato il programma di presentazione con il menù dei programmi. Digitate **RUN** seguito dalla pressione del tasto **RETURN**. Terminata la presentazione, per caricare uno qualsiasi dei programmi è sufficiente digitare:*
***LOAD "NOME PROGRAMMA"** seguito dalla pressione del tasto **RETURN**.*

Responsabile Editoriale Divisione Informatica
Francesca Marzotto

Direttore responsabile
Paolo Romani

Caporedattore
Marco Gussoni

Responsabile grafico Desktop Publishing
Adelio Barcella

Impaginazione elettronica
Mirko Diani, Giorgio Meroni

Collaboratori
Paola Bloise, Alessandra D'Ercole, Paolo Gussoni, Dolma Poli

Segretaria di redazione
Patrizia Angelo

Foto di copertina - Agenzia OLIMPIA

RadioELETTRONICA&COMPUTER
Rivista mensile, una copia L. 7.500, numeri arretrati il doppio del prezzo di copertina.
Pubblicazione mensile registrata presso il Tribunale di Milano n. 112 del 17/3/72.

Fotolito: Bassoli - Milano.

Stampa: GEMM Grafica srl, Paderno Dugnano (MI).

Diffusione: Concessionario esclusivo per l'Italia A.&G. Marco SpA, Via Fortezza 27 - 20126 Milano. Spedizione in abb. post. gruppo III/70.

Abbonamenti: Annuale L. 60.000, estero L. 80.000.

RadioELETTRONICA & COMPUTER è titolare in esclusiva per l'Italia dei testi e dei progetti di Radio Plans e Electronique Pratique, periodici del gruppo Societé Parisienne d'Edition.

Gruppo Editoriale JCE srl
Sede legale, Direzione, Redazione, Amministrazione
Via Ferri 6 - 20092 Cinisello Balsamo (MI)
Tel. 02/61.73.441 - 61.72.671 - 61.72.641- 61.80.228
Telex 352376 JCE MIL I - Telefax 61.27.620

Direzione Amministrativa
Walter Buzzavo

Pubblicità e Marketing
Gruppo Editoriale JCE - Divisione Pubblicità
Via Ferri 6 - 20092 Cinisello Balsamo (MI)
Tel. 02/61.20.586 - 61.27.827 - 61.23.397 - 61.29.0038

Concessionario eslusivo per Roma, Lazio e centro sud:
UNION MEDIA srl - Via G.B. Martini, 13
00198 Roma - Tel. 06/8119803-4
Telex 63026 UNION I - Telefax 06/5810537

Abbonamenti
Le richieste di informazioni sugli abbonamenti in corso si ricevono per telefono tutti i giorni lavorativi dalle ore 9 alle 12.
Tel. 02/61.72.671 - 61.80.228 - int. 311 - 338

Spedizioni
Daniela Radicchi

I versamenti vanno indirizzati a:
Gruppo Editoriale JCE srl, Via Ferri 6
20092 Cinisello Balsamo (MI), mediante l'emissione di assegno circolare, cartolina vaglia o utilizzando il c.c.p. n. 351205. Per i cambi di indirizzo allegare alla comunicazione l'importo di L. 3.000, anche in francobolli, e indicare insieme al nuovo anche il vecchio indirizzo.

Proprietario: *Editronica srl.*
Esercente l'impresa giornalistica:
GRUPPO EDITORIALE JCE srl.
Fanno parte del Gruppo Editoriale JCE srl:
JCE srl. - Editronica srl. - Edimedia srl.

Monitor Philips

Philips Consumer Electronics diversifica la sua produzione di monitor in due linee separate: Personal e Professional , entrando nel settore più sofisticato dei monitor Hi-end, distribuiti solo presso i negozi specializzati (**tavola 1**). La serie Personal, propone monitor a colori e monocromatici, da 14 pollici con entrate CVBS e RGB, compatibile con i più diffusi personal computer. I monitor della serie Professional sono studiati per soddisfare le esigenze più rigorose in quanto la loro tecnologia deriva dalla grande esperienza Philips nella costruzione di CRT, dalla qualità delle componenti costruite direttamente dalla casa madre e assemblati con un rigoroso programma di controllo qualità.

Commodisk 24

Come tutti i mesi Commodisk è in edicola con 5 fantastici programmi a 13 mila lire. Sul numero 24 grande spazio è dedicato alla prima parte dell'eccezionale integrato Mini Office II. L'intero pacchetto, che si completerà il mese successivo, è costituito da ben 6 programmi indipendenti, ma comunicanti.

Nella prima parte troverete il database, il programma di comunicazione e quello per disegnare i grafici, nella seconda il word processor, lo spreadsheet e il programma per stampare etichette. Il gioco del mese si chiama Scumball ed è un arcade classico, ma veramente divertente e pieno di sorprese.

Il quinto programma è come sempre un'utility. In particolare, Print-A-Sketch consente di disegnare sullo schermo con il joystick, in modo testo, e di salvare le schermate ottenute, in linee Basic, con le necessarie istruzioni Print per ottenere l'effetto disegnato. Il programma così ottenuto può essere unito a qualsiasi programma Basic per comporre intestazioni e pagine di presentazione.

Tavola 1. I monitor Philips

Modello	Prezzo e caratteristiche
Personal line (monocromatici)	
BM 7502	190.000 12" antiriflesso, CVBS fosfori verdi, ris. 920x300
BM 7522	190.000 12" antiriflesso, CVBS fosfori ambra, ris. 920x300
BM 7542	195.000 12" antiriflesso, CVBS fosfori bianchi, ris. 920x300
BM 7513	200.000 12" antiriflesso, TTL, fosfori verdi, ris. 920x350
BM 7913	250.000 12" flat square, TTL, fosfori verdi, ris. 920x350
BM 7923	250.000 12" flat square, TTL, fosfori ambra, ris. 920x350
BM 7752	250.000 12" flat square, TTL, fosfori bianchi, ris. 920x350
BM 7949	270.000 12" flat square, Analogico, fosfori bianchi, ris. 920x480
Personal line (a colori)	
CM 8802	530.000 14" CVBS e RGB, ris. 360x285
CM 8833	660.000 14" CVBS, RGB e RGB TTL, ris. 600x285
CM 9043	820.000 14" TTL, RGB, RGBI, EGA e CGA, 640x200 e 64x350
Professional line (monocromatici)	
7BM713	270.000 14" flat square, TTL, fosfori verdi, ris. 920x350
7BM723	270.000 14" flat square, TTL, fosfori ambra, ris. 920x350
7BM743	290.000 14" flat square, TTL, fosfori bianchi, ris. 920x350
7BM749	315.000 14" flat square, TTL, fosfori bianchi, ris. 920x480
Professional line (a colori)	
8CM852	750.000 14" black matrix, TTL RGBI, ris. 640x200
9CM053	900.000 14" black matrix, CGA/EGA, ris. 640x200, 640x350
9CM073	1.000.000 14" black matrix, CGA/EGA, ris. 640x200, 640x350
9CM082	1.050.000 14" black matrix, analogico, ris. 640x480
8CM875	1.300.000 14" CGA/EGA/PGA/VGA ris. 800x560

Le stampanti della nuova generazione

Un tempo solo una stampante Commodore poteva essere collegata direttamente alla porta seriale dei Commodore. Non c'era mol-

ta scelta, ma ora gli utenti del 64 e del 128 possono utilizzare le nuove stampanti ad aghi, a margherita e a trasferimento termico per risolvere le loro necessità di stampe in bianco e nero e a colori. Già dallo scorso anno la Okidata ha introdotto la stampante ad aghi Okidata 120, compatibile sia con la seriale Commodore sia con la parallela Centronics, utilizzabile da tutti i personal computer, dall'Amiga all'IBM.

Le stampanti ad aghi e a trasferimento termico sono le più popolari tra gli utenti Commodore. Gli ingredienti chiave della loro popolarità sono la versatilità e il basso costo. Comunque il prezzo non dovrebbe essere l'unico fattore determinante per la scelta di una stampante. Quelle ad aghi offrono velocità oltre alla possibilità di stampare complessi grafici.

Invece dei caratteri preformati, questa stampante utilizza una fila di aghi verticali che colpiscono la carta attraverso un nastro inchiostrato.

Quest'anno la Star Micronics America proporrà due versioni della stampante ad aghi NX-1000C. Entrambi i modelli hanno un dispositivo di inserimento dei fogli singoli utilizzabile senza rimuovere il trattore. La Star NX-1000C Multi Font ha quattro font selezionabili dal pannello di controllo frontale. La NX-1000C Rainbow possiede, oltre ai font interni multipli, sette possibilità di stampa a colori: nero, rosso, giallo, blu, arancio, verde e viola. La Rainbow può essere anche usata come stampante tradizionale per word processing e altro; infatti è sufficiente disinserire i comandi del colore e si otterrà una stampa in bianco e nero o monocromatica.

Prossimamente sarà distribuita in Italia anche la nuova nata della Okidata, la Okimate 20 che introduce una diversa tecnologia di stampa che utilizza il trasferimento termico. Infatti, invece di usare gli aghi sul nastro inchiostrato, scalda gli aghi che trasferiscono il calore al nastro imbevuto in un inchiostro simile a cera che sciogliendosi forma il carattere sulla carta.

Il procedimento a trasferimento termico ha diversi vantaggi confrontato con il metodo a margherita e ad aghi. Il più evidente è senz'altro la silenziosità, infatti produce un debole ronzio paragonato alle altre stampanti; ha inoltre un prezzo contenuto, ma la sua caratteristica più innovativa è il colore brillante delle sue stampe, che solo il trasferimento termico può raggiungere dal momento che le stampanti ad aghi trasferiscono il colore sulla carta con l'inchiostro che è privo di lucentezza. Sfortunatamente però consuma velocemente molto nastro oltre a richiedere carta molto liscia per ottenere buoni risultati.

Commodore Hospital

Aperto a Milano un "ambulatorio" per Commodore. Si chiama Computer Lab ed è un centro assistenza autorizzato per

prodotti Commodore con competenza regionale, si trova in Viale Monte Nero 66.

Alla consegna dell'apparecchio guasto viene rilasciato un tagliando di accettazione. Con questa carta d'identità, e con il programma Cat Master installato su un PC AT Commodore, verrà seguito in tutte le sue fasi, dalla diagnosi alle riparazioni.

Le riparazioni hanno una garanzia di tre mesi. Naturalmente, per i prodotti ancora coperti dalla garanzia Commodore l'intervento sarà gratuito. Per la gamma professional Computer Lab offre contratti di assistenza on-site e sta mettendo a punto l'installazione di Add-on su PC e Amiga anche se forniti dal cliente.

Computer Lab, Viale Montenero 66, Milano, telefono 02/5464436.

Super C

Una grandissima percentuale di possessori del C64 sfrutta il computer solamente con il software commerciale, mentre la maggioranza dei programmatori si dedica esclusivamente al Basic.

Questa scelta si basa su ottime motivazioni, infatti il Basic è un ottimo linguaggio per apprendere i fondamenti della programmazione, grazie alla sua intrinseca semplicità, inoltre è il linguaggio standard di tutti gli home computer. È d'altronde ovvio che un linguaggio così semplice su una macchina lenta come il C64 non permette certamente di ottenere buoni risultati con applicazioni che richiedano l'esecuzione di un grande numero di calcoli. Un parziale rimedio è costituito dai compilatori Basic, che sono in grado di velocizzare in misura considerevole alcuni tipi di programmi. Se però si desidera sfruttare al massimo le possibilità del C64 è necessario ricorrere al linguaggio macchina, cioè parlare la stessa lingua del microprocessore. Purtroppo il 6510 parla una lingua decisamente primitiva, infatti anche un'operazione semplicissima come per esempio un ciclo (la ripetizione di una serie di istruzioni) richiede un discreto numero di istruzioni, nonché una certa dimestichezza con la numerazione esadecimale e binaria.

Per ottenere i mirabolanti effetti grafici che possiamo ammirare in software commerciale, il linguaggio macchina è quasi indispensabile, tuttavia è necessario sapere che ha un temibile concorrente nel linguaggio C.

Il C non può essere inquadrato esattamente tra i linguaggi ad alto livello come il Basic, il Pascal o il Fortran, né può essere avvicinato al linguaggio macchina, che per definizione è il linguaggio a minore livello. Il C infatti raccoglie da una parte la semplicità d'uso dei linguaggi più evoluti, e dall'altra la possibilità di agire a fondo nella programmazione tipica del linguaggio macchina. A questo si deve aggiungere una grandissima modularità, ovvero la possibilità di programmare separatamente dei sottoprogrammi da utilizzare in seguito per diversi scopi. Nel nostro caso, ovvero per quanto riguarda la programmazione del C64, il C permette di ottenere risultati prossimi a quelli permessi dal linguaggio macchina, programmando in modo decisamente più elementare, più semplice e molto più visivo.

Questo lunghissimo preambolo per introdurre il compilatore Super C 64 della Abacus.

Questo compilatore è disponibile sia nella versione 2, utilizzabile con il Commodore 64 e con il Commodore 128, sia nella versione 3 riservata al solo Commodore 128. La versione 3 presenta considerevoli vantaggi: in particolare è possibile lavorare con lo schermo a 80 colonne, con la frequenza di clock raddoppiata e con un disco virtuale, creato sfruttando la maggiore disponibilità di RAM del C128, che velocizza enormemente i tempi di compilazione e di link. Il disco virtuale inoltre può addirittura essere utilizzato dai programmi, rendendo disponibile una notevole quantità di memoria.

Vediamo dunque com'è possibile iniziare a programmare in modo serio e diverso dal solito. Il pacchetto della Abacus è composto di un dischetto e di un sostanzioso manuale.

Il manuale consiste in un solido raccoglitore ad anelli nel quale sono racchiuse quasi 300 pagine di spiegazioni approfondite sull'uso del pacchetto software. La documentazione è davvero completa e ben realizzata, infatti oltre alla scontata descrizione del compilatore, dell'editor e del linker, troviamo un buon manuale di programmazione in C, una guida dettagliata alle librerie del sistema, nonché al particolare tipo di implementazione effettuata sul C64.

Passiamo ora a vedere il software vero e proprio. Il disco in dotazione è formattato su una sola faccia, e raccoglie un discreto numero di programmi esempio, oltre alle varie componenti del compilatore, alle librerie e i file.

La prima caratteristica rilevante di questo compilatore consiste nel fatto che utilizza un proprio sistema operativo, che in qualche modo emula l'ambiente Ms-Dos infatti, dopo aver attivato il programma caricando il primo file che si trova sul disco, compare sullo schermo una presentazione del sistema, nonché un prompt vistosamente simile al prompt Ms-Dos, ovvero la lettera a seguita dai due punti. Questo stadio del compilatore viene chiamato CCP, ossia stadio di comando C. Il prompt indica il drive attivo con la seguente convenzione: il drive a equivale al numero di device Commodore 8, il drive b al numero 9 e così via. È possibile collegare addirittura fino a otto drive. Per la versione 3 del programma esiste anche il drive h, che rappresenta il disco virtuale su RAM. È possibile cambiare il drive selezionato semplicemente digitando la lettera corrispondente al drive voluto.

Da CCP sono disponibili diversi comandi, quali:

- Dir - per vedere la directory del drive selezionato,
- Copy - per copiare un file,
- Type - per leggere un file testo,
- tutti i comandi del DOS standard Commodore, ovvero i vari Scratch, New, Validate, Initialize e così via.

Qualunque lavoro con il Super C inizia dal CCP. Per scrivere un proprio programma si deve utilizzare l'editor interno, chiamato ovviamente C Editor. Per caricare e fare partire l'editor è sufficiente digitare CE.

Questo editor è davvero ben fatto, e anch'esso ricorda molto l'ambiente di lavoro Ms-Dos, infatti è un editor decisamente professionale, se paragonato a programmi analoghi per C64. Con il C Editor è possibile utilizzare un numero di colonne variabile da 40 a 80. Naturalmente se si lavora su un C128 è scontata la scelta delle 80 colonne, mentre se si utilizza il C64, che non ne può mostrare più di 40, è necessario effettuare una scelta: lavorare con 40 colonne in C non è molto pratico, tuttavia se si seleziona una larghezza maggiore dello schermo si è costretti a subire lo scrolling orizzontale del testo che comunque è stato realizzato nel modo migliore.

I programmi devono essere salvati su un disco di lavoro, appositamente formattato. Il sistema operativo utilizzato dal Super C esce vistosamente dallo standard Commodore anche per le tecniche di registrazione, infatti tutti i file intermedi sono di tipo USR, ossia sono file gestiti direttamente dal computer, anziché dal floppy.

Questo sistema comunque presenta il vantaggio di una buona velocizzazione delle operazioni da disco. D'altra parte però non consente l'uso di alcun tipo di velocizzatore: neanche lo SpeedDos, che è probabilmente il velocizzatore più trasparente agli occhi del C64, può essere utilizzato con il Super C.

È decisamente utile, anche se non è assolutamente indispensabile, disporre di due drive, infatti in questo modo si evita di dovere continuamente sostituire il disco di lavoro con il disco di sistema. Naturalmente il secondo drive può essere anche virtuale, per chi dispone di un C128.

Dopo avere scritto un programma è necessario compilarlo, richiamando da CCP il compilatore, il quale, come la maggioranza dei compilatori C disponibili su sistemi ben più sofisticati del C64, si chiama CC, ossia C Compiler.

Il Compilatore richiede il nome del file da compilare, e restituisce il cosiddetto file oggetto, il quale non è ancora un programma eseguibile, poiché deve essere collegato ai file di libreria.

Per eseguire quest'ultima operazione si richiama il C linker, abbreviato CL, da CCP. Al linker bisogna specificare oltre al nome del file oggetto, il nome di tutte le librerie richieste. È inoltre disponibile un'opzione che permette di scegliere tra un file eseguibile da CCP e un file eseguibile dal basic standard del C64. Si sceglie la prima qualora si debbano fare delle prove, mentre quando il programma è terminato definitivamente si può selezionare la seconda, che produce un file caricabile dal C64 con la solita istruzione LOAD"nome",8. Naturalmente il vantaggio di quest'ultima opzione consiste nel fatto che per caricare un programma realizzato in tale modo, non è necessario caricare il Super C.

Passiamo ora a un commento sulle prestazioni offerte dal Super C. La velocità di compilazione e di linking è piuttosto buona, tenuto conto della lentezza congenita del drive del C64.

La dotazione di librerie è davvero buona: è possibile gestire in modo ottimale la grafica del C64, sprite compresi, e non

mancano neanche le funzioni matematiche. Il Super C permette davvero di realizzare programmi completi e di buon livello.

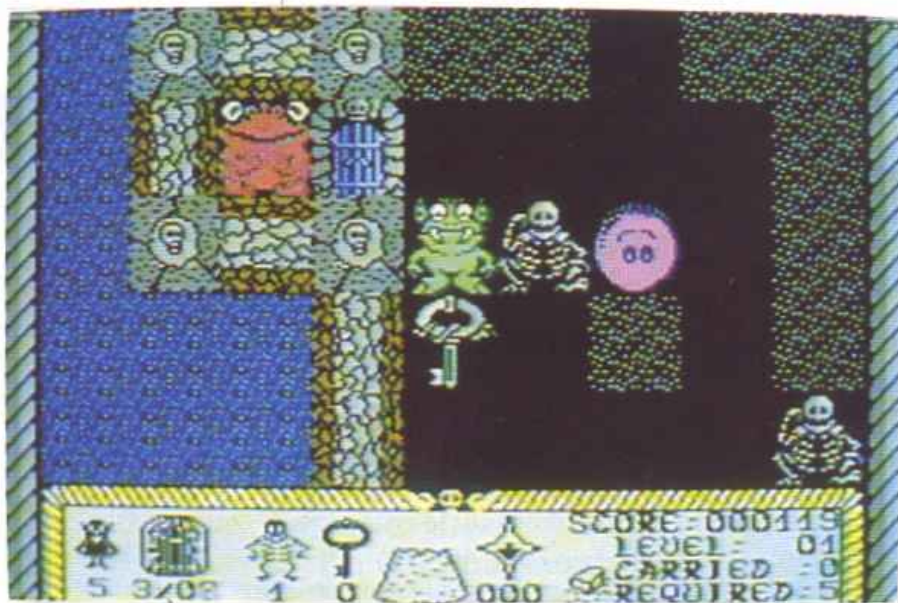
Per quanto riguarda la velocità di elaborazione il giudizio non può che essere positivo: certamente il Super C è il linguaggio che più si avvicina alla velocità del linguaggio macchina, e un adeguato sfruttamento delle librerie grafiche permette di realizzare persino buoni programmi che gestiscano in maniera molto efficiente il modo di alta risoluzione, nonché il multicolor.

Concludendo non possiamo far altro che consigliare vivamente questo compilatore a tutti gli appassionati di programmazione che desiderano lavorare in modo professionale, ma sufficientemente semplice, con il C64.

Videogames

Fra i numerosi videogames che animeranno prossimamente i vostri pomeriggi al computer ne abbiamo scelti alcuni che ci sembrano maggiormente degni di nota.

• Cominciamo con **Bone Cruncher**, un divertente arcade della Superior Software che si distingue per la sua originalità fra le miriadi di giochi uno uguale all'altro.

Il protagonista è un mostriciattolo che

passa la sua giornata a sgranocchiare le ossa degli scheletri che trova qua e là sul suo cammino. A rendergli la vita difficile ci pensano delle fastidiosissime palline rosa che non solo si combinano fra loro in modo da ostruire il passaggio, ma spesso e volentieri precipitano sulla testa del nostro povero amico mostro.

Scopo del gioco è fare tanti punti, ovvero divorare tante ossa e percorrere il maggior numero possibile dei cinque settori in cui è diviso ogni schema.

A movimentare il tutto contribuisce una grande varietà di particolarissimi esseri, alcuni amici del protagonista altri un po' meno.

La cosa più difficile da imparare in questo gioco consiste nel non farsi intrappolare o, peggio, intrappolarsi con le proprie mani disponendo gli elementi che compongono lo schermo di gioco in modo da non poterne più uscire.

In questo caso al nostro amico non rimarrà nient'altro da fare che...buttarsi in un pozzo!

Bone Cruncher è il tipico gioco che costringe il giocatore a riprovare innumerevoli volte, non tanto per migliorare il proprio record quanto per scoprire un pezzettino di schermo in più.

La grafica è buona e altrettanto si può dire della musica, ma attenzione: visto che probabilmente passerete molto tempo davanti a questo gioco vi consigliamo vivamente di escludere l'audio del vostro monitor prima di iniziare.

• Il secondo gioco che vi presentiamo è di tipo spaziale e si chiama **Fire Fly**. La Special FX che lo ha prodotto non va certo elogiata per la novità del soggetto; si tratta della solita navicella che vaga per i soliti spazi interstellari folgorando le solite innumerevoli astronavi aliene. Unica novità di rilievo è la schermata principale del gioco, una mappa stellare i cui quadranti vengono colorati superando la prova ad essi corrispondente.

Scopo del gioco è colorare caselle attigue in modo da collegare quella di partenza con quella di arrivo. Per evidenziare un quadrante basta passare su tutti i pianeti che compongono il settore di spazio al quale la casella si riferisce. Il problema è che per fare tappa su ognuno dei corpi celesti è necessario ingaggiare furiose battaglie con le astronavi che frequentano il posto, per cui non risulta sempre facile mantenere l'energia sufficiente per completare il quadro. Raggiunto un pianeta bisogna superare un'altra prova che varia di volta in volta per completare il settore.

La qualità della grafica varia da schermo a schermo ma si mantiene sempre a

un buon livello; la musica è raffinata ma vale il discorso fatto per il gioco precedente: fra laser, esplosioni e sottofondo musicale... vi consigliano di escludere l'audio.

• Cambiamo genere e occupiamoci di **Solomon's Key** creato dalla Tecno e distribuito dalla Probe Software.

Si tratta delle avventure di un intraprendente archeologo che, avendo profanato il sacro sepolcro del faraone, è costretto a subirne le conseguenze. La sua eplorazione nei labirinti che i sapienti architetti dell'antico Egitto hanno costruito è continuamente funestata da presenze a dir poco inospitali.

Oltretutto, ed è questo l'aspetto che più caratterizza il gioco, l'amico ricercatore deve crearsi egli stesso il percorso per uscire dalle diverse camere che compongono la piramide, sovrapponendo pietra su pietra in modo da costruire una sorta di scala e distruggendo i massi che gli ostruiscono il passaggio.

Insomma, come in tutte le cose, il successo finale va costruito "mattone su mattone"!

Anche se l'argomento del gioco può sembrare trito e ritrito bisogna sottolineare il fatto che a differenza degli altri videogames del genere in questo caso il labirinto viene creato dal giocatore stesso; per questo motivo a chi si cimenta in Solomon's Key non è richiesta solo abilità nel destreggiarsi col joystick ma anche una buona dose di ragionamento per disporre opportunamente le pietre.

Un particolare apprezzabile del programma consiste nella legenda che appare prima di iniziare a giocare; essa suddivide i personaggi che si incontrano durante il gioco in "buoni" e "cattivi", cioè quelli che, toccandoli, portano punti e quelli che invece fanno perdere una vita. Questa idea, benché semplicissima, permette al novizio di non giocare tutte quelle inutili partite per capire chi bisogna evitare e chi non.

Per finire alcuni giudizi in generale: non si può dire che l'ideatore del gioco si sia sforzato eccessivamente per inventare la musica che appare piuttosto elementare; anche la grafica non è eccezionale ma in questo tipo di gioco non è estremamente importante.

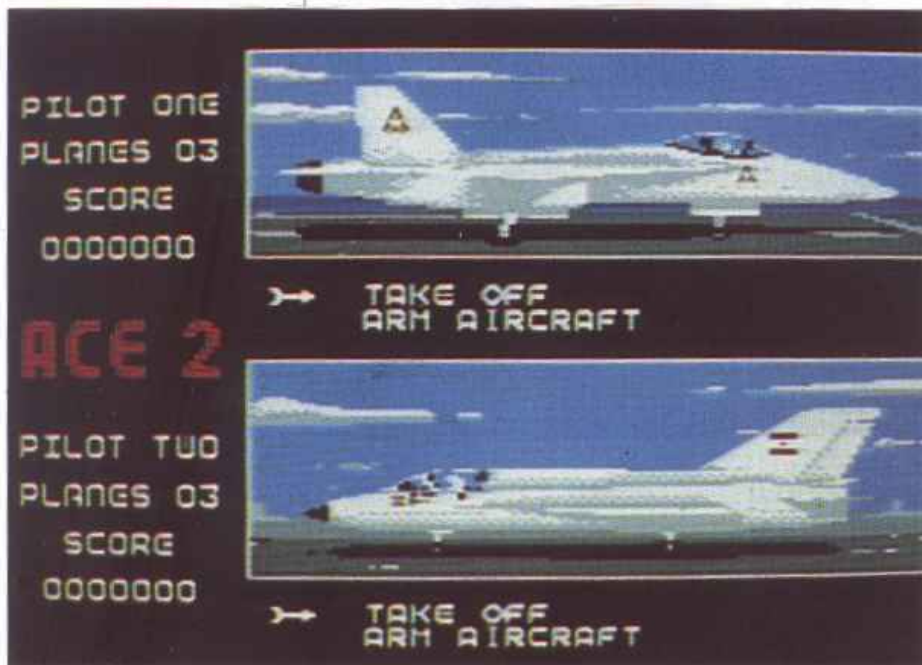
• Dall'antico Egitto passiamo ai giorni nostri o quasi ed entriamo in uno scenario da guerra fredda.

Stiamo parlando di **Jagd Auf Roter Oktober**, una simulazione di guerra navale in cui una volta tanto siamo al comando di un sottomarino russo al quale è stato dato l'ordine di compiere un'incursione nei mari occidentali. Districarsi fra mine vaganti, navi nemiche e amiche, strumenti di bordo e armamenti vari è tutt'altro che facile, ma siamo sicuri che una volta assunta un po' di dimestichezza col gioco non mancheranno i motivi per appassionarsi.

Non si può non rilevare la precisione con cui è riportato ogni dettaglio riguardante la conduzione di un sottomarino; anche i dialoghi fra comandante ed equipaggio che appaiono su un apposito display contribuiscono notevolmente al realismo.

Unica pecca, inevitabile se si vuole ricostruire uno scenario più vero possibile, è

rappresentata dalla difficoltà in cui ci si trova nel scegliere i comandi, tutti effettuabili col joystick ma non sempre immediati. Originale la trovata di far conoscere l'esito della missione tramite le pagine di un giornale.

Ultime due note per il singolare cursore a forma di falce e martello e per lo scrolling della presentazione iniziale veramente notevole.

Particolare non trascurabile è il fatto che il gioco è scritto completamente in tedesco: tenete quindi un vocabolario sottomano.

• Ritornando ai giochi di casa nostra

parliamo di **Bocce**, creato dalla Simulmondo per la Italvideo.

Come avrete capito si tratta di una produzione completamente made in Italy anche se non si capisce il motivo per cui le scritte debbano essere necessariamente in inglese.

Apprezzabile dal punto di vista grafico la presentazione iniziale ma tanto dispendio di energie poteva essere utilizzato per completare il gioco vero e proprio che in realtà appare un po' scarno.

Sarebbe sicuramente stata più utile una vista laterale del campo di gioco e soprattutto una maggior cura nel rendere verosimili gli urti fra le bocce.

È specialmente questa ultima caratteristica che colpisce maggiormente in negativo: scagliare a tutta forza una boccia e vedere che quella urtata si muove come fosse spostata da un alito di vento non fa sicuramente una bella impressione!

Bocce può essere giocato da due persone o singolarmente contro il computer che, anche ai livelli più bassi, è pressoché imbattibile.

Al momento del lancio il giocatore può scegliere la posizione di partenza, la direzione del tiro, l'effetto che la pallina assumerà dopo aver toccato terra.

Infine, manovrando opportunamente il joystick, si può graduare la potenza del tiro. Grafica discreta, musica assente.

• Non è tale da destare troppo interesse **Ace 2** della Cascade Games, un incolore proseguimento di Ace.

Si tratta di una battaglia aerea nei cieli di una non meglio precisata regione della Terra.

I contendenti devono inseguirsi e abbattersi fino all'esaurimento del carburante e anche oltre se riescono ad atterrare per rifornirsi.

La grafica è migliorata, perlomeno nella presentazione e, a differenza della versione precedente, il combattimento può avvenire non solo contro il computer (insuperabile se non dopo ore e ore di addestramento) ma anche fra due giocatori.

Al pilota sono richieste né più né meno le stesse prestazioni di qualsiasi gioco del genere: evitare lo stallo, non arrischiarsi in picchiate troppo audaci, tenere costantemente sotto controllo la quota di volo, non farsi accodare dall'avversario e sparare non appena si ha nel mirino il veivolo nemico.

Insomma Ace 2, come Rambo 2, Superman 3 e Rocky 4, è destinato a scomparire fra qualche tempo senza lasciare alcuna traccia di sé.

• 23 ottobre 1944: il comando generale

della sesta flotta vi ordina di garantire la sicurezza della navigazione nella zona di mare a sud delle Midway; i giapponesi pur essendo a pochi passi dalla sconfitta potrebbero operare gli ultimi tentativi di rovesciare la situazione.

Non è la sceneggiatura di "La battaglia delle Midway" bensì lo scenario di **Power at See**, ideato dalla Accolade.

Quella sopra citata è una delle missioni nella quale il giocatore, ammiraglio di una portaerei americana, dovrà destreggiarsi nel migliore dei modi.

Gestire il comando di una delle più grandi navi che abbiano mai solcato i mari comporta una grande responsabilità: da voi dipende l'incolumità di migliaia di uomini e di decine di aeroplani, oltre naturalmente il destino della patria.

Per questo dovrete essere pronti a contrastare le incursioni aeree nemiche non appena l'ufficiale addetto al rilevamento radar ve ne darà notizia; dovrete ordinare raid preventivi sulle basi nemiche disponendo di bombardieri perfetti per lo scopo; dovrete far partire gli stormi di caccia dalla tolda della nave per intercettare gli aerei giapponesi prima che raggiungano la vostra flotta.

In questo arduo compito sarete aiutati dagli ufficiali che vi terranno costantemente informati sulle condizioni delle navi e dall'addetto alla radio che vi collegherà direttamente al comando supremo per saperne di più sull'evolversi della situazione.

Inoltre potrete avere sotto gli occhi una mappa dettagliata della zona delle operazioni per scegliere meglio la tattica da usare.

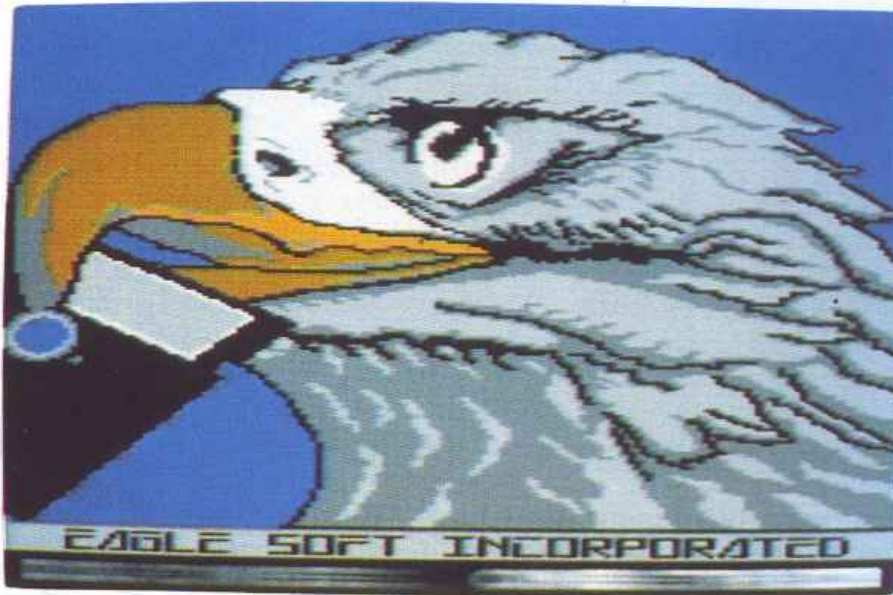
Nel complesso Power at See è un wargame di ottima fattura e riesce ad appassionare il giocatore anche non essendo ricchissimo d'azione.

È apprezzabile l'idea di far scegliere al giocatore stesso la quantità di uomini, aerei, combustibile con cui iniziare la missione; ovviamente queste quantità sono bilanciate direttamente dal computer, cioè maggiore è il numero di caccia prescelti, minore sarà quello dei bombardieri e la stessa cosa vale per le truppe e il carburante.

In questo modo il potenziale offensivo iniziale è uguale ogni volta.

Buona la grafica come pure la musica che si sposa bene con la solennità dell'argomento.

• E infine **Manic Mansion**, una produzione Lucas Film. Si tratta di un adventure che però, a differenza degli altri, non si articola in un monotono susseguirsi di disegni: questa volta i protagonisti sono animati e si muovono sullo schermo di gioco come i personaggi di un normale arcade.

Novità anche per quel che riguarda la selezione dei comandi da dare: essi sono scritti tutti in basso allo schermo, così non si pone più il problema di doversi inventare ogni volta la parola giusta per una determinata operazione.

Anche il nome degli oggetti appare automaticamente sullo schermo ponendo il cursore su di essi e premendo Fire.

La storia da cui prende spunto l'adventure è presto raccontata: tre ragazzi devono salvare una loro amica che un terribile scienziato pazzo tiene rinchiusa nella sua tetra villa.

Egli vuole impossessarsi del cervello dell'ostaggio per compiere i suoi inumani esperimenti.

Riusciranno i nostri eroi a liberare la ragazza dalle sue grinfie? Ce la faranno a raggiungere il laboratorio dello scienziato senza incontrare la temibile infermiera alleata del professore?

A voi la risposta, o meglio, a voi il joystick.

La grafica del gioco è abbastanza elementare, specialmente per quel che riguarda le figure animate, ma non poteva essere diversamente.

Ottima la presentazione, una vera e propria sigla di testa con tanto di musica da thriller e di panoramica sulla notte di luna piena in cui si ambienta il tutto.

Gianni Arioli

Un ottimo strumento didattico per migliorare e verificare la vostra conoscenza del Basic. Quattro programmi che vi sottoporranno una fitta serie di problemi di programmazione ispirati a situazioni reali.

Basic Test

Il modo più spontaneo e stimolante per misurare la propria conoscenza di un linguaggio di programmazione sarebbe quello di essere interrogati direttamente dal computer. Basic Test è un vero e proprio supporto didattico completissimo che vi permetterà di verificare, con l'aiuto del vostro C64, la vostra conoscenza del Basic. Non si tratta del solito corso visivo in cui, con una serie di schermate più o meno numerose, viene riassunto velocemente il linguaggio, ma di un vero e proprio corso interattivo.

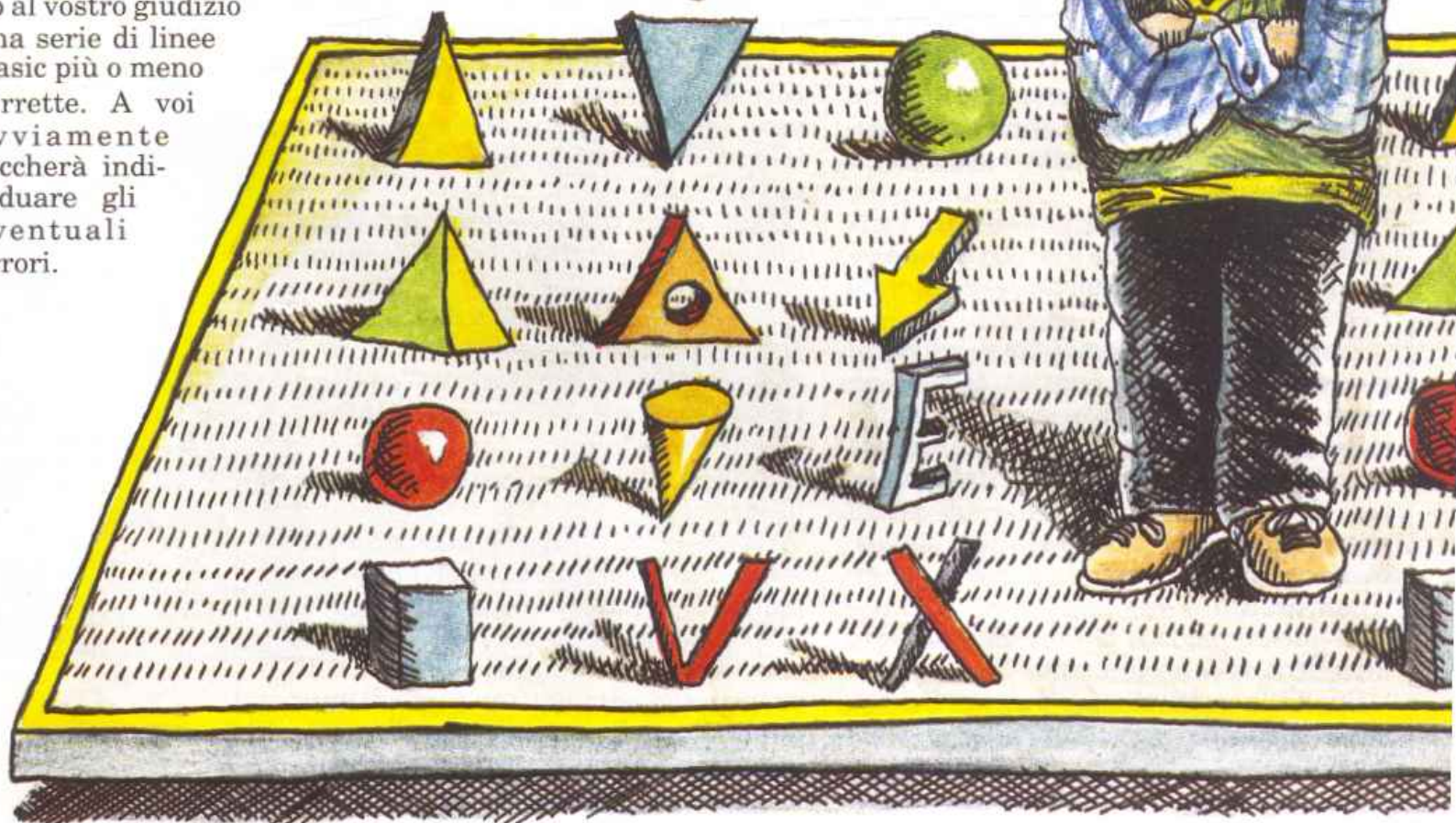
Il computer si trasformerà in un docente severissimo ma al tempo stesso paziente ed estremamente "umano" sottoponendo al vostro giudizio una serie di linee Basic più o meno corrette. A voi ovviamente toccherà individuare gli eventuali errori.

Come funziona

Basic Test è articolato in 4 programmi indipendenti che realizzano altrettante interrogazioni sul linguaggio. Sulla cassetta allegata alla rivista è registrato il primo di questi programmi, Basic Test 1. Per caricarlo è sufficiente digitare in modo diretto l'istruzione: Load"basic test 1", seguita da Return. Una volta caricato in memoria potete lanciarlo con il consueto Run.

I programmi sono tutti in Basic e si prestano anche a eventuali miglioramenti perché hanno una struttura molto semplice.

Una volta lanciato il programma si può scegliere se vedere le istruzioni, premendo il tasto F1, oppure iniziare subito il test, premendo la barra spazio. Le istruzioni riguardano le modalità da seguire per selezionare la risposta e altre infor-

mazioni di carattere generale sulla struttura del test.

Ogni programma deve essere visto come una unità indipendente dal momento che la suddivisione non è fatta per argomenti ma per grado di difficoltà. Ovviamente gli argomenti più complicati come la gestione del drive verranno trattati soltanto nelle ultime interrogazioni.

I quattro programmi condividono la stessa struttura.

Ogni programma sottopone al "videostudente" una serie di 10 domande alle quali si può rispondere senza alcun limite di tempo.

Più che di domande si tratta di linee Basic che devono essere studiate e quindi giudicate in base alle regole sintattiche elencate sul manuale di programmazione.

Ogni linea Basic visualizzata deve essere digitata in modo diretto, cioè priva di un numero di linea che la precede.

Inoltre, tutte le linee di programma corrette devono essere considerate come realmente eseguite dal computer.

Ciò significa che per rispondere sulla correttezza di una linea Basic si deve tener presente anche l'effetto delle precedenti linee Basic corrette.

Per esprimere il proprio giudizio sulle linee di programma si deve utilizzare la barra spaziatrice con cui è possibile selezionare una fra le tre seguenti risposte: valido, non valido e dipende. La risposta che lampeggia è quella selezionata e per confermare la scelta si deve premere il tasto Return. Dopo ogni vostra risposta il computer provvederà a visualizzare la soluzione corretta del quesito fornendo sia la risposta che andava selezionata sia una breve spiegazione sull'effetto della linea Basic da esaminare.

Ovviamente si deve sempre fare molta attenzione alla spiegazione perché solo le linee indicate come corrette saranno eseguite dal computer e avranno effetto sulle domande successive.

La soluzione corretta viene visualizzata immediatamente sotto la linea Basic mentre la spiegazione della linea di programma compare in campo rosso a metà schermo.

Nella parte inferiore dello schermo vengono indicati il numero di risposte corrette date e la valutazione della vostra soluzione. Al termine delle 10 domande viene formulato il giudizio globale sulla prova effettuata.

In questa fase il C 64 si rivela di manica abbastanza larga dando la sufficienza anche a chi risponde correttamente a sole 4 domande.

Quindi il test riprende dall'inizio ripetendo nello stesso ordine le medesime domande.

Un giudizio

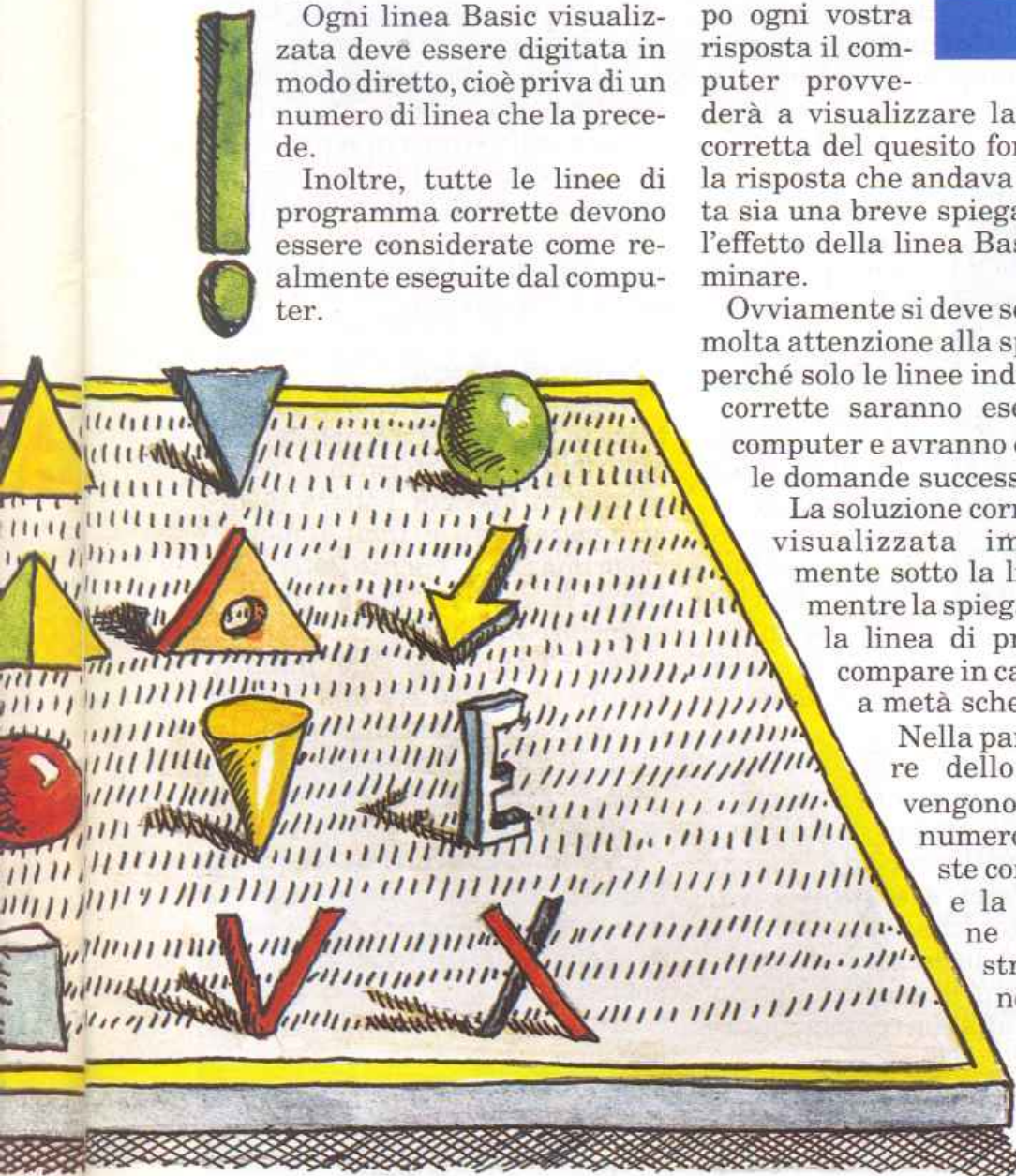
La grafica risulta molto ben curata senza però cadere nell'eccesso e anche gli effetti sonori che accompagnano la valutazione delle risposte e che aprono l'interrogazione sono gradevoli.

Anche le domande, pur non coprendo tutti gli aspetti del C64, sono sviluppate secondo un ordine logico molto rigoroso e ricostruiscono con estrema fedeltà le situazioni reali in cui è possibile trovarsi durante la programmazione effettiva.

La modalità di valutazione ricalca fedelmente i canoni di una normalissima prova scolastica, scostandosi da questi ultimi solo per la formulazione del giudizio finale.

Ogni risposta esatta dà diritto a un punto e per avere la sufficienza basta raggiungere 4 punti. Tenete presente però che la sufficienza non è un giudizio positivo ma soltanto un incoraggiamento a fare di meglio.

Francesco Milione

(continua)

Salta in groppa al tuo scooter spaziale, distruggi le sfere robot e affronta i tuoi nemici in questa avventura mozzafiato.

Dan Dare II

Il nostro personaggio principale è un agente speciale con un incarico del tutto particolare. Nella flotta di astronavi che aveva il compito di perlustrare un pianeta molto strano, il tenente Peter aveva il compito di guidare la squadriglia da guerra.

Per un disastroso incidente dovuto alla sua negligenza, ci fu una violenta esplosione con conseguenze mortali per tutto l'equipaggio.

Solo alcune persone, tra cui il tenente, si salvarono.

Ovviamente al ritorno sulla Terra fu esonerato dall'incarico e processato da una corte marziale che lo accusò addirittura di assassinio volontario.

Condannato all'ergastolo riuscì però a fuggire con l'aiuto di pochi amici fidati. Con la sua astronave partì per lo spazio lasciando una lettera al comando, sulla quale scrisse che si sarebbe vendicato nella maniera più atroce.

Sono passati molti anni da quel tempo e nessuno seppe più niente del tenente Peter.

Finché un brutto giorno, sul radar della base a terra del secondo corpo d'armata del Pacifico, non apparve un puntino misterioso.

Partirono subito dei caccia per mettersi in contatto con il misterioso oggetto volante.

Dei due caccia non si seppe più nulla e, quello che era peggio, il veicolo si stava avvicinando sempre più alla zona dei vulcani nel Pacifico.

Nessun contatto, nessuna comunicazione, niente... finché arrivò il momento della trasmissione... Era proprio lui, il tenente scomparso nello spazio e considerato disperso che diceva di essere tornato sulla terra per vendicarsi e per distruggerla.

Ovviamente, i lunghi anni trascorsi nello spazio da solo avevano cancellato anche quel poco di razionalità rimasta in lui e lo avevano fatto diventare completamente pazzo.

Aveva infatti costruito una squadriglia di robot (Super treen), su delle strane motociclette che avevano il compito di sbarcare sulla Terra e scaricare delle bombe protoniche nelle bocche dei vulcani così da creare una rivoluzione a catena in grado di riportare la Terra alla sua condizione primordiale.

Tutto era studiato nei minimi particolari ma, se un individuo fosse riuscito a penetrare all'interno dell'astronave, non si sa come avrebbero reagito i robot che non avevano mai visto prima un essere umano oltre al tenente Peter. Il comando militare provò ad attaccare l'astronave con tutte le armi a disposizione, missili, caccia, bombardieri... tutto sembrava inutile.

Si pensò, infine, come ultima speranza, di mandare una specie di Rambo, il colonnello Dan, detto Danny.

Armato di sola pistola laser e a cavallo di uno scooter spaziale si avvicinò all'astronave senza essere intercettato dal radar nemico. Dopo varie ore di lavoro riuscì, con la pistola laser, ad aprirsi un varco sul fondo e a penetrare all'interno della navetta.

Come si gioca

Appena all'interno, ecco che entrate in gioco voi. Lo scopo del gioco è infatti quello di trovare i robot avvolti in speciali bolle e di distruggere i loro sistemi di memoria prima che il tenente Peter li attivi e li spedisca sulla Terra a sganciare le bombe.

Per riuscire nel suo intento, Danny deve sparare alla scatola di controllo di ciascuna bolla. In basso al centro dello schermo è segnato il tempo a disposizione: solo 10 minuti.

Una volta distrutti tutti i robot, il colonnello deve pensare a salvare la sua vita prima che l'astronave esploda e deve dirigersi verso le porte blindate, che riconoscerete facilmente, e restare lì fino all'esplosione.

Se non riuscirete a distruggere tutti i robot girovaghi o ad arrivare fino alle porte entro il tempo

prefissato il gioco finisce. Al contrario se si riesce a portare a termine la missione si passa al livello successivo. Il nostro eroe ha solo sei vite a disposizione e l'energia per ogni vita è segnalata in basso e decresce ogni volta che Danny viene colpito: esaurita una barretta Danny perde una vita.

La grafica è di ottima qualità sia per gli sprite sia per i fondali, precisi e ricchi di colori vivaci. L'unica pecca sono gli effetti sonori, alquanto miseri per un gioco così ben congegnato. La presentazione è molto simpatica e l'azione del gioco è abbastanza frenetica e vi darà la sensazione, purtroppo vera, che il tempo a disposizione sia poco.

Sono consentiti pochi errori e ogni tunnel imboccato a caso vi costerà caro.

Non si può dire che questo gioco abbia la sua carta vincente nelle opzioni anche se ne ha una alquanto interessante. Si può scegliere infatti se pilotare il buono o il cattivo. Se pilotate Danny, sapete già cosa fare; se pilotate il tenente Peter invece, avrete il compito di attivare manualmente le scatole di controllo delle bolle. Il tempo a disposizione è sempre di 10 minuti in quanto il colonnello Danny ha predisposto un timer per l'autodistruzione dell'astronave.

Per cominciare inserite il joystick e premete Fire. In basso sullo schermo sono visualizzati il punteggio e i messaggi mandati dal computer di bordo.

Consigli

Un consiglio che possiamo darvi è quello di disegnare su un pezzo di carta una mappa dell'astronave in modo da non sbagliare il percorso e sarebbe meglio farlo per ognuno dei quattro livelli.

L'impresa per portarli a termine tutti non sarà facile e dovrete scoprire un bel po' di enigmi e aspettarvi simpatiche sorprese lungo il tragitto.

Non scoraggiatevi alle prime difficoltà perché, una volta capito il meccanismo, questo fantastico gioco vi darà mille soddisfazioni.

Ricordatevi che la calma è la virtù dei forti anche nei momenti nei quali vi sentite spacciati. In alcuni momenti i robot non sembrano troppo svegli e girano a caso: approfittate di questi attimi. Le feroci sparatorie contro gli inesauribili Super treen non mancheranno e anche i più bravi tra voi non avranno vita facile.

Oscar Maeran

Per salvare su disco i programmi di RE&C

Per usare Dsave va specificato: il nome del file su disco, l'indirizzo iniziale dell'area di memoria e l'indirizzo finale della stessa. Ogni indirizzo richiede 2 byte in notazione byte basso - alto. Non sempre è necessario specificare gli indirizzi, perché il programma Dsave ha i suoi valori di default, che sono quelli più comuni per i file della cassetta. Per procedere al salvataggio di un programma su disco occorre seguire i seguenti passi: **1.** Pulite lo schermo con SHIFT-HOME, posizionate il cursore sulla quarta riga dall'alto e impostate il set di caratteri minuscolo. **2.** Caricate il programma Dsave con LOAD (se da disco con LOAD"DSAVE",8,1). Appariranno 3 righe di caratteri, di cui la più bassa con il messaggio "dsave attivato". La routine in LM è nella memoria video, quindi non cancellate mai lo schermo. **3.** Caricate da cassetta il programma che desiderate salvare su disco con LOAD e attendete fino a quando non ricompare il cursore. **4.** Spostate il cursore sulla seconda riga dello schermo e modificate opportunamente il nome del file che appare verso la fine della riga. Per default troverete <NOME FILE>, potete modificare questo nome sovrascrivendo i caratteri dal simbolo "<" al simbolo ">" usando caratteri shiftati. E' necessario usare caratteri shiftati perchè nella memoria video i caratteri sono in codice POKE, mentre la routine che utilizza il nome per il SAVE li considera in codice ASCII. Quando avete finito di scrivere il nome del file, senza battere RETURN, riportate il cursore in basso a inizio riga. Se dovete salvare programmi che partono dall'area Basic seguite i passi 5a e 7. Altrimenti seguite i passi 5b e 6.

5a. Se dovete salvare un programma che parte dall'area Basic, quindi con indirizzo iniziale 2049 che corrisponde al default, non dovete impostarlo. L'indirizzo finale è letto automaticamente dai puntatori 45 e 46 e anche questo non va impostato in quanto viene settato automaticamente al momento del caricamento da cassetta. Non resta che dare il comando di attivazione: SYS1024. Il programma è stato salvato. Saltate al punto 7. **5b.** Dovete salvare un segmento di memoria, per esempio un programma in l.m. il cui indirizzo iniziale è diverso da 2049. In questo caso l'indirizzo iniziale è sempre indicato nell'articolo che descrive il programma. Supponiamo che l'indirizzo iniziale sia 28576. Eseguite quindi le seguenti istruzioni per settare il puntatore di inizio: POKE1089,INT(28576/256):POKE1088,28576-256*INT(28576/256). **6.** L'indirizzo finale è già stato settato dal caricamento da cassetta e quindi non vi resta che dare la SYS1024.

7. Se volete salvare un altro programma riportate il cursore sulla quarta riga di schermo e ripartite dal punto 3. Per salvare il programma Dsave su disco, infine, è necessario caricare in memoria Dsave da cassetta seguendo i punti da 1 a 3; quindi impostate i parametri di inizio e fine e salvate con le seguenti istruzioni: POKE1088,0:POKE1089,4:POKE45,120:POKE46,4:SYS1024.

Il computer è infallibile? Vedremo insieme che non è affatto così. Naturalmente sono errori piccoli, ma è importante valutarli e, se necessario, correggerli.

Gli errori del computer

L'idea largamente diffusa che il computer sia infallibile, almeno per quanto riguarda i calcoli matematici, è dura da sfatare. L'idea mitica del cervello elettronico infinitamente potente e preciso si scontra tuttavia con la dura realtà sperimentale. Intendiamoci, gli errori di cui stiamo parlando sono insignificanti se rapportati alle quantità che siamo abituati a maneggiare nella vita di tutti i giorni, ma diventano rilevanti quando si cerca di risolvere numericamente problemi complessi, che richiedono, come noto, moltissime operazioni iterate. Posto il problema è necessario anche tentare una soluzione. Vedremo a questo proposito alcuni accorgimenti empirici, che, anche se non hanno una solida giustificazione teorica, si rivelano vincenti nella pratica. Digitate in modo diretto:

```
PRINT7.832-7.064
PRINTSQR(616225)
PRINT785^2
```

Le risposte del computer saranno rispettivamente:

.767999997 anziché .768
785.000001 anziché 785
616225.001 anziché 616225

Provate con:N=60:M= 60*60: PRINT SQR(M)-N

A video apparirà 1.49011612E-08 anziché 0. In quest'ultima prova il computer mostra di non riuscire nemmeno a capire che 60 e la radice quadrata del suo quadrato sono la stessa cosa. Che succede, allora, se in un ciclo di calcoli spuntano questi errori? Succede che si trascinano per tutta la sequenza delle operazioni, si sommano ad altri errori e alla fine si avrà un risultato imperfetto che, se anche sarà sufficientemente approssimato, tuttavia non potrà accontentare il matematico pignolo, il quale, se non è un pignolo, è un matematico. Il difetto più grave del computer è quello di non riuscire spesso a riconoscere due entità uguali, e ciò pone dei problemi, per esempio, quando con IF... THEN... si chiede un confronto deve essere operata una scelta.Poiché i dati prima immessi sono ancora in memoria, chiediamogli:

```
IF SQR(M)<>N THEN PRINT"E' DIVERSO".
```

Sbagliando, il computer risponderà “E' DIVERSO”. Supponiamo ora di dover sottrarre dalla radice quadrata di 710649 un numero qualsiasi, per esempio 5. Poiché la radice quadrata di 710649 è 843, come risultato dovremmo ottenere 838. Invece per il computer non è così. Digitiamo, sempre in modo diretto il comando: PRINT(710649)-5. Avremo come risposta 838.000001: Perché? Perché se battiamo PRINT SQR (710649), verrà stampato 843.000001 anziché 843. A questo punto viene da chiederci, se gli errori del computer siano frequenti. Mettiamolo alla prova.

Ordiniamogli di elevare tutti i numeri da 1 a 1000 al quadrato, di estrarre e mostrare la radice quadrata del quadrato ottenuto e, infine, di sottrarre dalla radice quadrata il numero iniziale, visualizzando la risposta che, ovviamente dovrebbe sempre essere zero. Perciò diamo il New e scriviamo:

```
10 FORA=1TO1000
20 K-A*A
30 Kl=SQR (K): PRINT "; Kl =";Kl
90 PRINT"Kl-A ="; Kl-A
100 PRINT
110 FORTM=1TO500:NEXT
120 NEXT
```

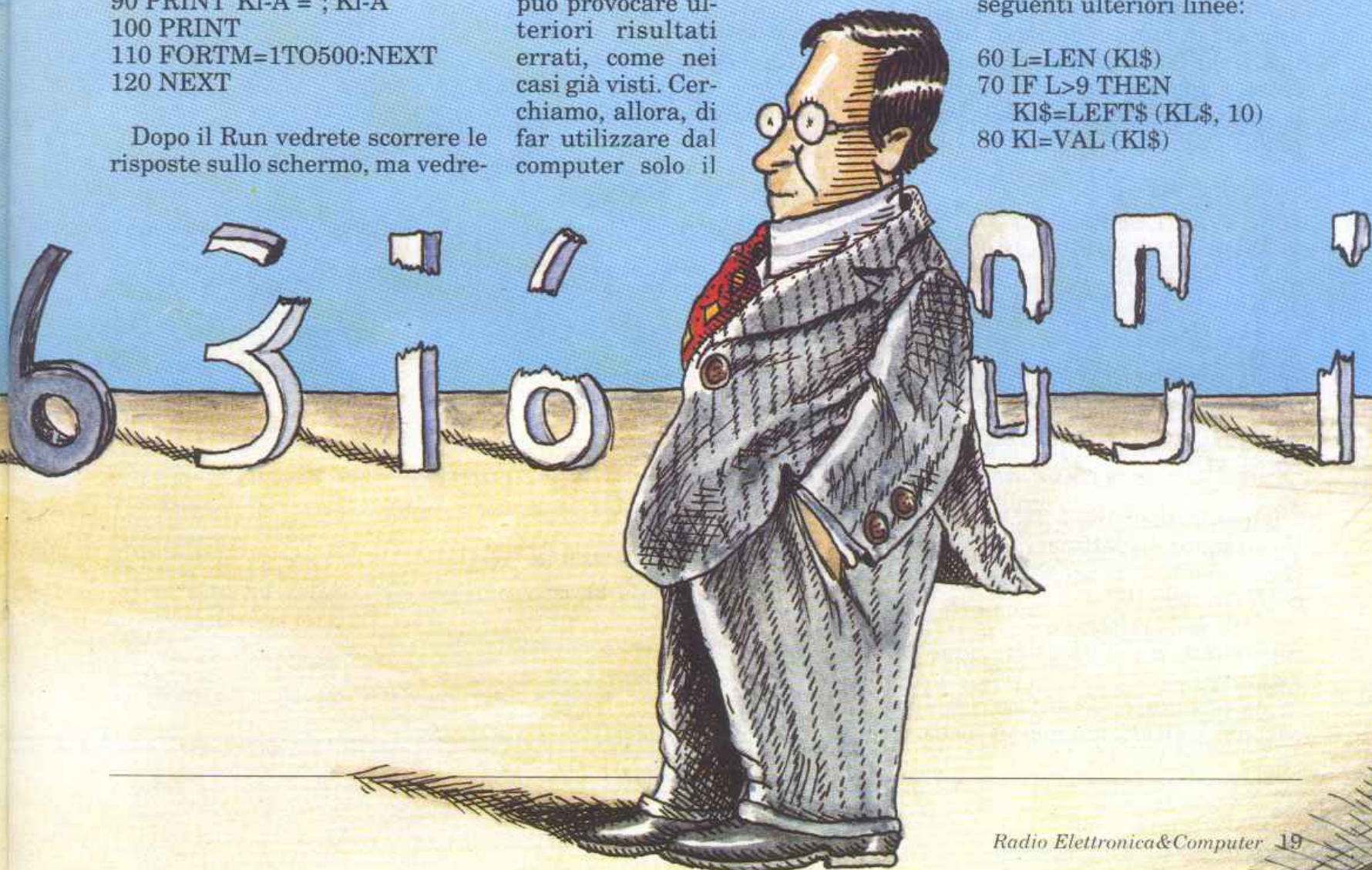
Dopo il Run vedrete scorrere le risposte sullo schermo, ma vedrete anche che Kl-A sarà uguale a un numero espresso in notazione scientifica di valore inferiore a .0000001. Noteremo anche che:

- Il valore di Kl (ossia la radice quadrata del quadrato di A) indicato dal computer, che dovrebbe sempre essere uguale ad A, alle volte è uguale, ma spesso è diverso? Quando è diverso il suo valore è pari a quello di A aumentato di .0000001 per A da 1 a 99 (tranne per A=163) e di .000001 per A da 100 a 999.
- Se il computer attribuisce a Kl-A un valore diverso da zero, lo fa sia quando giustamente visualizza per Kl lo stesso valore di A, sia quando erroneamente ne visualizza uno discordante da A.

Ne deriva dalla seconda osservazione che anche nei casi in cui la radice quadrata che appare sul video è esatta, il valore contenuto nella variabile Kl che la rappresenta è diverso nella memoria del computer. Utilizzando la variabile per un'altra operazione, il computer non tiene conto del valore visualizzato, ma di quello che ha in memoria, e ciò può provocare ulteriori risultati errati, come nei casi già visti. Cerchiamo, allora, di far utilizzare dal computer solo il valore visualizzato e non quello in memoria. Inseriamo perciò nel programma le seguenti linee:

```
40 Kl$=STR (Kl)
50 Kl=VAL (Kl$)
```

Con il ricorso alle funzioni stringa VAL e STR$, al Run il computer comincerà a funzionare meglio, perché darà risposta giusta all'operazione di sottrazione in corrispondenza di ogni radice quadrata (Kl) indicata esattamente. Naturalmente il computer continua a sbagliare il risultato della sottrazione là dove è già sbagliata la radice quadrata evidenziata. Abbiamo visto che la radice quadrata di 710649 segnalata dal computer è 843.000001 invece di 843. Dobbiamo eliminare l'uno finale in questo e in tutti i casi analoghi. Per farlo, toglieremo dalla stringa Kl$ (ottenuta con la linea 40) il suo ultimo carattere per ricavare poi il valore della stringa così depurata. Raggiungeremo tale scopo con l'uso delle funzioni stringa LEN e LEFT$, ed inserendo nel programma le seguenti ulteriori linee:

```
60 L=LEN (Kl$)
70 IF L>9 THEN
   Kl$=LEFT$ (KL$, 10)
80 Kl=VAL (Kl$)
```

Fatto questo, vedremo che le risposte che scorrono sullo schermo sono tutte esatte, anche dove la radice quadrata visualizzata è errata. Si ha eccezione per A=163 (*). Rimedieremo, rettificando la linea 30 con:

```
30 Kl=SQR (K*10000)/100:
   PRINT"Kl =";Kl
```

Finalmente ogni risposta sarà 0. Dall'insieme degli accorgimenti introdotti nel programma che abbiamo costruito, si può ricavare la seguente formuletta:

```
SQR(K) = VAL(LEFT$
         (STR$(SQR(K*10000)
         /100),10))  (**)
```

che permette:
- di depurare le radici quadrate dalle approssimazioni del computer, se le radici quadrate devono essere numeri interi;
- di lasciare inalterate le radici quadrate rappresentate da numeri decimali limitati, se le cifre, fra interi e decimali, non sono più di otto.
- di ridurre le radici decimali illimitate in numeri decimali limitati, il che lascia pressoché inalterate le approssimazioni. Vogliamo controllare l'efficacia della formula? Diamo il New e digitiamo:

```
10 FORK-1TO10000
20 Kl=VAL(LEFT$(STR$
   (SQR(K*10000)/100),10))
30 PRINT "SQR(";K;") = ";Kl
40 Kl%=Kl:IFKl%*Kl%-K=0
   THEN GOSUB60
50 NEXT
60 PRINTK; "QUADRATO DEL
   NUMERO INTERO:"; Kl
70 FORTM=0TO1000: NEXT
80 RETURN
```

* Il quadrato di 163 è 26569.
Il computer visualizza:
SQR (25659) = 162.999976
SQR (25659)-163 =
-2.45571137E-05
(fornendo un'approssimazione negativa)
** da utilizzare, beninteso, per i casi qui trattati, ma non in sosti-

Listato 1. Equazioni di secondo grado.

```
10 rem"equaz.sec.grado"
15 poke53281,7
20 printchr$(144)
30 print"[clr]"
40 printtab(7)"[rev on][2 down]equazioni di secondo grado"
50 printtab(10)"[down]*******************"
60 print"[down][right]l'equazione da risolvere deve essere"
70 print"[down][right]preventivamente ridotta alla forma"
80 printtab(10)"[2 down][rev on]a*x^2 + b*x + c = 0"
90 print"[2 down][right]i coefficienti [rev on]a[roff], [rev on]b[roff], [rev
   on]c[roff] possono[32]essere:"
100 print"[down][right]-numeri interi"
110 print"[right]-numeri decimali"
120 print"[right]-frazioni"
130 printtab(12)"[3 down](premi f1)"
140 getk$:ifk$<>"[f1]"then140
150 print"[clr]"
180 print"[2 down][3 right]se le soluzioni, o radici, sono rap="
190 print"[down][2 right] presentate da numeri decimali, ver="
200 print"[down][2 right]ranno visualizzate anche le loro"
210 print"[down][2 right]frazioni generatrici ridotte ai mi="
220 print"[down][2 right]nimi termini, tranne quando il di="
230 print"[down][2 right]scriminante e' negativo 0 superiore"
240 print"[down][2 right]a 999999999, oppure se la sua radice"
242 print"[down][2 right]quadrata e' un numero decimale illi="
245 print"[down][2 right]mitato."
250 printtab(13)"[down][right](premi f1)"
260 getk$:ifk$=""then260
270 print"[clr]":print"[down][right]imposta i coefficienti dell'equazione"
280 print"[right]da risolvere:"
290 input"[2 down][right]primo coefficiente  :";a$
300 input"[down][right]secondo coefficiente:";b$
310 input"[down][right]terzo coefficiente  :";c$
320 printtab(10)"[down]* * * * * * * * *"
330 print
340 gosub2200
350 print"[down]l'equazione da risolvere e' la seguente:"
360 print
370 printtab(1)a$" * x^2"k1$b$" * x"k2$c$" = 0"
380 printtab(10)"[3 down](esatto ? s/n)"
390 getk$:ifk$="n"then270
400 ifk$<>"s"then390
410 rem trasforma le variabili stringa
415 rem in variabili numeriche e consi=
420 rem dera frazionari tutti i coeffi=
425 rem cienti, assegnando '1' come de=
426 rem nominatore a quelli non frazio=
427 rem nari.
430 d$=a$:gosub2000
440 ad=d:a3=d3:a5=d/d3:rem a5 utilizzata in linea 1090
450 d$=b$:gosub2000
460 bd=d:b3=d3
470 d$=c$:gosub2000
480 cd=d:c3=d3
490 rem ricerca il min. comune multiplo
495 rem fra i denominatori.
500 m=a3:n=b3
510 gosub2110
520 m=y:n=c3
530 gosub2110
540 rem elimina i denominatori
545 rem effettua: num.*m.c.m./den.
550 a4=ad*y/a3
560 b4=bd*y/b3
570 c4=cd*y/c3
580 print"[clr]"
590 rem calcola il discriminante
600 d1=val(str$(b4*b4*10000))
610 d2=val(str$(4*a4*c4*10000))
620 d=val(str$((d1-d2)/10000))
630 ifd<0thenprinttab(5)"[down][right][rev on]il discriminante e'
    negativo[roff]":goto650
640 ifa4=0andb4=0thenprint"[3 down][2 right][rev on]non risolvibile-manca
    l'incognita ![roff]":goto2500
650 ifa4=0thenprint"[2 down][right]equazione di primo grado con una sola
    radice:":goto670
660 goto680
670 print"[down][right]x = ";-c4/b4:goto2500
680 ifb4=0andd<0then1130
690 ifb4=0andc4=0thenprint"[down][right]equazione incompleta con unica
    radice:":goto710
700 goto730
710 print"[down][right]x = 0":goto2500
720 gosub2300
730 ifb4=0thenprinttab(13)"equazione pura"
735 ifb4=0thenprint"[down][right]d (discriminante) =";d:print"[down][right]le radici
    sono:":goto1170
```

```
740 ifc4=0thenprint"[down][right]equazione spuria - le radici sono:":goto1080
750 ifd<0thenprint"[down][right]d (discriminante) =";d
760 ifd<0thenprint"[down][right]radici complesse coniugate":goto1010
770 gosub2600
850 x1=m/p:x2=n/p
860 print"[down][right]equazione trinomia - le radici sono:"
870 print"[2 down][right]x1 = ";x1;"; x2 = "; x2
875 print"[down][right]——"
876 print"[down][right]d (discriminante) =";d
880 x1%=x1:x2%=x2:rem da 1088 e 1180
890 k1=len(str$(x1%)):k=len(str$(x1))
900 j2=len(str$(x2%)):j=len(str$(x2))
910 ifk1=kandj2=jthen2500
912 ifd<999999999then916
913 ifd>999999999thenprint"[2 down][right]non effettuo la ricerca delle
    frazioni"
914 print"[down][right]ridotte ai minimi termini generatrici"
915 print"[down][right]delle radici, perche' d > 999999999.":goto2500
916 rem se sqr(d) e' decimale finito,
917 rem cerca le generatrici,altrimenti
918 rem informa di non essere in grado.
920 k=len(str$(h))
930 ifk>9then950
935 ifk<=9andval(str$(h*h))<>dthen950
940 goto1190
950 print"[2 down][right]non si ricercano le frazioni ridotte ai"
960 print"[right]minimi termini,generatrici delle radi="
970 print"[down][right]ci,perche' la rad.quadrata del discri="
980 print"[down][right]minante e' un n.ro decimale illimitato."
1000 goto2500
1010 d=-d:rem goto 1010 in 760
1020 r=-b4/(2*a4)
1030 f=sqr(d):f$=str$(f):f=val(f$)/(2*a4)
1040 print"[down][right]x1 = :";r;" + (";f;"*i)"
1050 print"[down][right]x2 = :";r;" - (";f;"*i)"
1060 gosub2400
1070 goto2500
1080 gosub2600:rem goto 1080 in 740
1084 x1=m/p:x2=n/p:print"[down][right]x1 =";x1;" ;  x2 =";x2
1085 print"[down][right]d (discriminante) =";d
1086 ifn=0thenn=m
1087 ifm=0thenm=n
1088 goto880
1090 ifa5<0then1110:rem il valore di a5 e' in 440
1100 print"[down][right]x1 = +";sqr(a5);"; x2 = -";sqr(a5):goto2500
1110 print"[down][right]x1 = +";sqr(abs(a5));"* i; x2 = -";sqr(abs(a5));"* i"
1120 gosub2400
1125 goto2500
1129 rem goto 1130 in 680
1130 print"[2 down][right]d (discriminante) =";d:printtab(10)"[down]———"
1135 print"[down][right]equazione pura con radici immaginarie"
1140 d=-d:f=sqr(d):f$=str$(f):f=val(f$)/(2*a4)
1150 print"[down][right]x1 =";f;"* i"
1155 print"[down][right]x2 =";-f;"* i"
1160 gosub2400
1165 goto2500
1170 gosub2600:rem goto 1170 in 735
1180 x1=m/p:x2=n/p:print"[down][right]x1 =";x1;" ; x2 =";x2:goto880
1190 rem cerca il m.c.d.-goto1190 in940
1195 rem tra 'm', 'n' e 'p'.
1200 r1=sgn(x1):r2=sgn(x2)
1210 t=p
1220 ifsgn(m)=-1thenm=-m
1230 ifsgn(p)=-1thenp=-p
1240 m=val(str$(m))
1250 p=val(str$(p))
1260 u=m:j=p
1270 h=m:i=p
1280 ifm>pthen1310
1290 ifm<pthen1320
1295 rem x=m.c.d fra 'm' e 'p'
1300 x=h*i/m:goto1330
1310 p=p+i:goto1280
1320 m=m+h:goto1280
1330 ifsgn(n)=-1thenn=-n
1340 ifsgn(t)=-1thent=-t
1350 v=n:p=t
1360 n=val(str$(n))
1370 p=val(str$(p))
1380 h=n:i=p
1390 ifn>pthen1420
1400 ifn<pthen1430
1405 rem y=m.c.d fra 'n' e 'p'
1410 y=h*i/n:goto1450
1420 p=p+i:goto1390
1430 n=n+h:goto1390
1440 rem usa il m.c.d. per trovare le
```

Continua a pag. 22.

tuzione, per esempio, di SQR (K)*SQR (K).

Avviato il Run, vedrete scorrere le radici quadrate dei numeri da 1 a 10000 prive di approssimazioni quando devono essere rappresentate da numeri interi, e constaterete anche che il computer ora riconosce i quadrati dei numeri interi, cosa che non gli sarebbe riuscita di fare senza l'aiuto della suddetta formula (o di altra equivalente). Vediamo un altro esempio di utilizzo delle predette funzioni stringa. Digitiamo in modo diretto:

M=0.06*1000:N=60:PRINT"M =";"; M-N = "M-N

Al Return il computer visualizzerà:

M =- 60 ; M-N = 1.49011612E-08

che è errato perché dovrebbe rispondere: M-N = 0. Il computer, eseguita la moltiplicazione, visualizza il risultato esatto, ma ha in memoria un valore approssimato che influisce sulla successiva operazione di sottrazione. Accertiamoci di tutto ciò, chiedendogli ancora una volta, in modo diretto: PRINTM confermerà: 60. Eppure, forse M è diverso da 60? Proviamo:

IFM<>60 THEN PRINT "SI"

Il computer risponde Sì. Anche qui, per rimediare, basta usare VAL e STR$. Perciò digitiamo (i dati prima introdotti sono sempre in memoria):

PRINT"M-N=";VAL(STR$(M))-N

ecco che ora risponde: M-N = 0, che è quanto dovevamo attenderci. Errori del computer si verificano anche nell'elevamento a potenza con il segno, al quale è preferibile l'uso del segno*, se l'esponente non è talmente elevato da costringere ad un eccessivo lavoro.

Digitiamo, tenendo presente che 23^409 è il quadrato di 153:

M-1532:N=23409:PRINT"M-N =";M-N

Avremo:

M-N = 7.62939453E-06

Invece, con:

PRINT"M-N=";VAL(STR$(M))-N

uscirà M-N = 0, vale a dire, è stata eliminata l'approssimazione

Segue da pag. 21.

```
1445 rem frazioni generatrici
1450 print"[2 down][2 right]le radici espresse sotto"
1460 print"[down][2 right]forma frazionaria sono:"
1465 ifr1*u=0then1480
1470 print"[down][right]x1 = ";r1*u/x;"/";j/x
1475 ifr2*v=0then2500
1480 print"[down][right]x2 = ";r2*v/y;"/";j/y
1490 goto2500
1990 rem gosub2000 nelle linee 450 e 470
2000 w=len(d$):q=asc(d$):ifq=45thend$=right$(d$,w-1)
2010 d=val(d$):d1$=str$(d):l=len(d$):m=len(d1$):s=l-m
2020 ifs<1thend3=1:goto2040
2030 d2$=right$(d$,s):d3=val(d2$)
2040 ifq=45thend=-d
2050 return
2100 rem gosub 2110 nelle linee 510 e 530
2110 ma=m:na=n
2120 ifm>nthen2150
2130 ifm<nthen2160
2140 y=m
2141 return
2150 n=n+na:goto2120
2160 m=m+ma:goto2120
2200 k1=sgn(val(b$)):k2=sgn(val(c$))
2210 ifk1=1ork1=0thenk1$=" +"
2220 ifk2=1ork2=0thenk2$=" +"
2230 ifk1=-1thenk1$=chr$(32)
2240 ifk2=-1thenk2$=chr$(32)
2250 return
2290 rem gosub 2300 in 720
2300 ifd>0then2310
2310 print"[down][right]radice quadrata"
2320 print"[right]del discriminante =";sqr(d)
2330 return
2390 rem gosub 2400 in 1060,1120,1160
2400 printtab(28)"[down]rr"
2410 print"[right]i = unita' immaginaria = mn-1"
2420 return
2490 rem gosub 2500 nelle linee670,710,910,915,1000,1070,1125,1165,1475,1490
2500 print"[2 down](premi [rev on]f1[roff] per continuare, [rev on]f3[roff]
     per finire)"
2510 geta$:ifa$<>"[f1]"anda$<>"[f3]"then2510
2520 ifa$="[f3]"then2540
2530 ifa$="[f1]"then270
2540 end2590 rem eliminazione approssimazioni
2600 h=val(left$(str$(sqr(d*10000)/100),10))
2650 m=val(str$(-b4*1000))+val(str$(h*1000))
2660 n=val(str$(-b4*1000))-val(str$(h*1000))
2670 p=val(str$(2*a4*1000))
2680 return
```

nell'operazione di elevamento a potenza. Abbiamo visto all'inizio che il computer può sbagliare anche una semplice sottrazione fra numeri decimali non elaborati, che non siano cioè il risultato di precedenti operazioni.

Quando ci sono di mezzo i decimali, bisogna avere la paziente accortezza di moltiplicarli per 10, per 100, per 1000 e così via (cioè per 10 con esponente sufficiente a eliminare i decimali) e alla fine dell'operazione in cui essi siano coinvolti dividere il risultato per la potenza di 10 usata in precedenza.

Naturalmente non va dimenticato il solito ricorso alle funzioni stringa VAL e STR$. Esempio:

- con PRINT 7.832-7.064, esce .767999997 mentre con PRINT (VAL(STR$(7.832*1000))-VAL(STR$(7.064*1000)))/100 si ha .768

- con PRINT (7.832-7.064)*3 esce 2.30399999 mentre con PRINT ((VAL (STR$ (7.832* 1000))) -VAL(STR$(7064*1000)))/1000)*3 si ha 2.304, che è il risultato esatto.

Finiamo qui la caccia agli errori. Quelli trovati sono sufficienti per permetterci di sperimentare l'efficacia dei rimedi escogitati.

Facciamolo affrontando la formula risolutiva dell'equazione di 2° grado. Malgrado la sua apparente semplicità, permette al computer di infilarvi tutte le imprecisioni di cui è capace. È quindi un banco di prova per i nostri accorgimenti. Il programma al riguardo elaborato risulta piuttosto esteso perché analizza le diverse specie dell'equazione di 2° grado.

Inoltre, oltre a coefficienti interi o decimali, il programma accetta anche coefficienti frazionari, e questo costringe a inserire tutti i coefficienti, interi, decimali o frazionari, sotto forma di stringa, per poi trasformarli in variabili numeriche. La presenza di frazioni comporta che tutti i coefficienti siano considerati tali, e pertanto il programma assegna come denominatore il valore di 1 ai coefficienti interi o decimali.

Deve poi eliminare i denominatori delle frazioni, calcolandone il minimo comune multiplo. Eseguite le operazioni previste dalla formula, se le radici sono rappresentate da numeri decimali, visualizza anche la loro frazione generatrice ridotta ai minimi termini (purché il discriminante non sia superiore a 999999999 o la sua radice quadrata non sia un numero decimale illimitato). Per far questo deve trovare anche il massimo comun divisore tra "-b +√b2-4 ac", "-b -√b2 -4 ac" e "2a". Se il discriminante è negativo, evidenzia I = unità immaginaria = √-1.

Scorrendo il listato, riscontrerete l'impiego dei vari accorgimenti di cui abbiamo parlato. I commenti nelle Rem intendono essere di sufficiente aiuto alla comprensione dello svolgersi del programma.

Provate a risolvere gli esercizi proposti da un testo d'algebra sul quale siano anche riportate le soluzioni. Salvo errori di stampa sul testo, non dovreste mai trovare discordanze (nel senso che le approssimazioni del computer sono eliminate) fra le risposte del testo e quelle del computer.

Inventatevi equazioni strampalate. Il programma le affronterà con successo, purché non inseriate coefficienti tali da costringere a cercare m.c.m. e m.c.d. fra numeri in notazione scientifica.

Se inserite coefficenti frazionari e per ottenere le frazioni generatrici delle radici, il computer qualche volta potrebbe darvi l'impressione di essersi incantato. Invece, lavora solo come un matto. Aspettate con pazienza!!

Gustavo Fontanella

Il tema è quello classico dei primi videogames: il muro di mattoni da distruggere; ma la grande varietà di livelli e la perfezione grafica lo collocano tra gli arcade più irresistibili.

Fuga da Zart

Malgrado a prima vista sembri una versione riveduta e corretta di Break out, un gioco uscito molti anni fa, non ha nulla a che vedere con quest'ultimo!

È infatti un gioco irresistibile, dotato di buoni effetti sonori, grafici, e di movimento.

È ambientato in un'epoca del futuro dove i criminali, per scontare le loro pene, vengono mandati in un particolare sistema solare e rinchiusi in un particolare labirinto sul pianeta Zart. Ignari di tutto ciò, degli esploratori spaziali che girovagavano in lungo e in largo per l'universo, decisero di esplorare anche il pianeta Zart, contro il volere della federazione spaziale.

Con la loro astronave si portarono in orbita attorno al pianeta e cominciarono ad esaminarlo con particolari sonde. Aria respirabile, cibo in abbondanza, lussureggiante vegetazione... era così bello da sembrare irreale.

Decisero quindi di scendere di persona per vedere se gli strumenti non fossero impazziti, ignorando così l'avvertimento della federazione spaziale che più volte aveva cercato di convincere l'equipaggio a non scendere su quell'infernale pianeta.

Purtroppo per loro, una volta atterrati avrebbero trovato una terribile sorpresa: innanzitutto le trasmissioni si erano istantaneamente interrotte e il motore della navicella sembrava bloccato da una forza invisibile.

Dopo poco, colti da uno strano presentimento, videro sotto i loro occhi il pianeta che si stava trasformando. In breve tempo assunse il suo vero, terribile e misterioso aspetto. Uno di loro gridò: "Era tutta un'illusione!!"; presi dal panico cercarono di fuggire, ma troppo tardi! Un misterioso raggio li colpì facendo perdere loro conoscenza.

Mentre il pianeta riprendeva il suo aspetto fantastico e illusorio si risvegliarono in un'universo contrario. Tutto era rovesciato e opposto alla normalità.

Sospinti da una forza misteriosa si ritrovarono in uno strano labirinto. Sul portone d'ingresso non riuscirono a scorgere l'insegna sulla quale era scritto il luogo dove erano capitati: Traz (Zona ad Alto Rischio Traumatico).

È questo infatti il posto dove venivano esiliati i delinquenti.

Il labirinto è una sorta di prigione composta da 64 celle. In ognuna di esse gli

sventurati sono costretti a sostenere prove infernali.

Con delle particolari racchette devono distruggere tutti i blocchetti di energia facendoli scontrare con una pallina. Sta ora a voi impersonare i nostri impavidi esploratori cercando di far loro riconquistare la libertà.

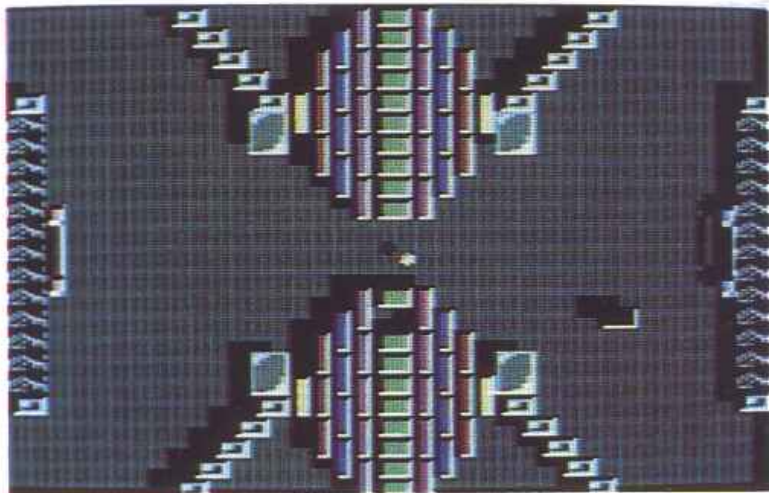
Come si gioca

Dopo una presentazione degna di questo gioco, appare lo schermo delle opzioni.

Qui, potete scegliere tra uno o due giocatori e un particolare constrution kit che poi vedremo a che cosa serve. Appena ini-

ziate la partita vi trovate in uno schermo a caso del labirinto.

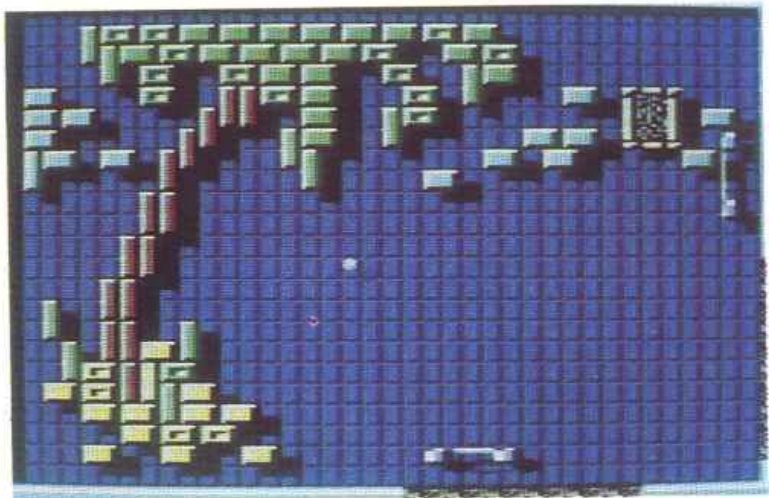
Facendo rimbalzare la pallina sulla racchetta, dovete cancellare tutti i pezzettini colorati dallo schermo. In alcuni schermi (quasi tutti) vi sono due racchette.

Se sono poste in orizzontale si potranno comandare con il joystick muovendolo orizzontalmente a destra e sinistra, se invece sono poste verticalmente, in alto e in

basso. La loro velocità può venire aumentata se premete contemporaneamente il tasto del joystick.

Ogni quadro contiene un lato del muro perimetrale di forma discontinua (porta).

Se la pallina va a cadere lì, perderete una vita.

Altri ostacoli sono mattoncini indistruttibili, trappole per la pallina, specchi rifrangenti eccetera.

Alcuni cubetti sono dei generatori di mostri: infatti se vengono colpiti creano

delle immmagini (facce, bocche, occhi) che corrono per lo schermo e fanno deviare la pallina se la toccano.

Alcune volte si vede passare sullo schermo un punto di domanda. Se riuscite a prenderlo con la racchetta, crea un effetto particolare al gioco.

Per la maggior parte delle volte è benefico, vi regala una vita, vi fa finire il quadro... ma alcune volte è dannoso in quanto fa uno strano effetto alla pallina che si mette a girare vorticosamente senza una traiettoria rettilinea, la fa andare più veloce, la fa diventare invisibile...

C'è anche la possibilità che vi capiti la possibilità di sparare: premendo il bottone del joystick la vostra racchetta sparerà e ogni mattoncino verrà distrutto rapidamente.

Finito il quadro, ammesso che ci riusciate vista la difficoltà di alcuni quadri, appaiono sullo schermo delle frecce.

Queste indicano la direzione da seguire all'interno del labirinto.

Se porterete a termine tutti e 64 i livelli sarete riusciti a far scappare i nostri esploratori. In ogni momento è possibile ispezionare la mappa e la posizione attuale del labirinto.

Alcuni schermi sono davvere difficili e presentano addirittuta quattro porte, quattro racchette e nove palline!

Schermi ben disegnati, un realistico movimento della pallina, effetti sonori un po' strani fanno di Traz qualcosa di formidabile. L'azione del gioco non viene neanche rallentata quando vi sono venti sprite in campo.

L'opzione del construction kit è formidabile: consente infatti di cambiare tutti i parametri del gioco e di creare, come uno vuole, i singoli quadri del labirinto.

Consigli e conclusioni

Se avete scelto l'opzione a due giocatori, dovrete stare particolarmente attenti a due cose: 1) se fate cadere una pallina nelle porte, entrambi i giocatori perderanno una vita; 2) attenzione al cambio di racchetta.

Bisogna inoltre, mentre si gioca, essere molto attenti al sonoro.

Se sentite un rumore insolito bisogna fare molta attenzione perché, visto che i due giocatori giocano contemporaneamente, potrebbero scambiare tra loro le racchette creando panico, confusione e, purtroppo, perdendo una vita.

Non scoraggiatevi se all'inizio non riuscirete a completare tutti e 64 gli schermi, ci vuole molta pratica e sangue freddo. Buon divertimento!!

Oscar Maeran

Anche il drive 1541 cambia look. La nuova versione del glorioso disk drive per Commodore 64 si presenta più elegante, compatta e con un prezzo ancora più accessibile.

Commodore 1541-II

La differenza fondamentale tra la cassetta e il floppy disk consiste nella possibilità di quest'ultimo di accedere direttamente a qualunque dato, mentre il registratore può solamente leggere i dati che si trovano disposti lungo il nastro uno dopo l'altro cesso. In sostanza il floppy disk risparmia la faticosissima ricerca del punto iniziale della registrazione di un programma, poiché nella directory contiene l'elenco completo dei file, pertanto, senza l'intervento dell'utente che deve solo fornirne il nome, cerca i dati richiesti e sposta la testina di lettura al punto opportuno. In linea di principio inoltre il disk drive è molto più ve-

loce di un registratore, tuttavia questo non era vero per il C 64; infatti per il registratore Commodore, che funzionava con una lentezza davvero esasperante, è stato creato il famosissimo Turbo Tape che ha incrementato la velocità di caricamento di ben dieci volte, con il risultato di una situazione paradossale del nastro leggermente più veloce del disco. In breve tempo poi è stato creato il primo Turbo Disk, al quale ne sono seguiti molti altri, e allora il disk drive ha iniziato davvero a prendere il sopravvento sul registratore, con il risultato che attualmente la maggioranza degli utenti Commodore dispone di questo comodissimo accessorio. Oggi, a qualche anno dalla nascita del primo drive per Commodore 64, il glorioso 1541, la Commodore ha presentato l'erede, il 1541-II. In realtà sarebbe più corretto parlare di fratello minore, infatti le differenze tra i due modelli sono davvero poche.

L'estetica è ovviamente ispirata al nuovo Commodore 64, infatti il colore è identico e anche la forma delle prese d'aria ricorda il suo predecessore. A differenza degli altri modelli di drive (il 1540, che era destinato al Vic20, ma funzionava anche con il Commodore 64, nonché il 1570 e il 1571 per il C128, anch'essi compatibili con il 1541) l'alimentatore è esterno. Il principale vantaggio consiste nel fatto che il drive vero e proprio non si surriscalda: gli utenti del 1541 sanno benissimo che dopo alcune ore di utilizzo intenso del loro drive la temperatura inizia a diventare preoccupante. Anche l'alimentatore del 1541-II si surriscalda, tuttavia il corpo-drive si mantiene a temperatura ambiente anche dopo alcune ore di uso intenso.

Un altro piccolo vantaggio, più estetico che altro, consiste nella minore dimensione della macchina, sicuramente più maneggevole ed elegante.

C'è però il rovescio della medaglia: l'interruttore di accensione si trova situato sul corpo drive, quindi a valle dell'alimentatore anziché a monte, pertanto agendo solo su tale comando il trasformatore resta sotto tensione e l'intero alimentatore assorbe una seppur minima quantità di corrente anche a drive spento. Perché non è il caso di lasciare un alimentatore collegato alla tensione di rete anche quando non viene utilizzato, si rende necessario staccare la spina ogni volta che si spegne il drive. Poiché il 1541-II è sostanzialmente identico al 1541, è evidente che quasi tutti questi dati sono noti a chi possiede già un drive Commodore. Cominciamo con qualche informazione di natura circuitale. Il 1541-II non è un semplice floppy disk drive con il relativo controller, ma è un vero e proprio computer, infatti dispone di un microprocessore (un 6502, già utilizzato dal Vic20, nonché dall'Apple II), di memoria ROM (ben 16 Kb di sistema operativo), di 2 Kb di memoria RAM utilizzati per i buffer, per le variabili di sistema e per eventuali programmazioni da parte dell'utente, nonché da due 6522 (chiamati VIA o Versatile Interface Adapter) che servono per interfacciare la CPU con l'elettronica esterna, ossia i controlli del motore e della testina, nonché il bus dati per comunicare con il computer.

Quando il Commodore 64 accede al disk drive ha luogo una vera e propria comunicazione tra computer. Un vantaggio offerto da tale sistema è costituito dal fatto che se per esempio si vuole formattare un disco, basta dare il comando al drive, il quale provvede a eseguire l'ordine, mentre il computer esegue altre istruzioni. Proseguiamo con un dato fondamentale: la capacità. Il 1541-II può accedere a una sola facciata per volta del floppy disk , infatti ha una sola testina, pertanto per floppy disk intenderemo una singola facciata, restando inteso che se il disco è a doppia faccia è possibile voltarlo per sfruttare ambedue i lati. Il 1541 formatta i dischi a 35 tracce, ciascuna delle quali contiene da 17 a 21 settori (contraria-

mente alla maggioranza dei sistemi operativi che hanno un numero fisso di settori per traccia, il DOS Commodore dispone di un maggior numero di settori sulle tracce esterne). In realtà il 1541-II è in grado di accedere fino alla quarantesima traccia, tuttavia per compiere questa operazione deve essere programmato appositamente. Questo meccanismo richiede un'approfondita conoscenza del sistema operativo, ed è spesso utilizzato per la protezione dei dischi. La traccia 18 è utilizzata per la directory, mentre i blocchi liberi per l'utente sono 664, per un totale di quasi 169.000 bytes. Poiché la lunghezza media di un buon programma per il Commodore 64 varia dai 30Kb ai 60Kb, possiamo vedere che in un floppy è possibile registrare da tre a sei programmi. Il 1541 ammette quattro tipi di file, ossia di gruppi di dati. Il tipo PRG è il più comune, e in tale modo sono solitamente registrati i programmi. Un file PRG consiste in un insieme di bytes e in un indirizzo. L'indirizzo si riferisce naturalmente alla RAM del computer, ed è esattamente l'indirizzo iniziale del gruppo di bytes. Un file di tipo PRG può essere caricato con l'istruzione LOAD"nome",8, nel qual caso il caricamento avviene a partire dalla locazione $0801 indipendentemente dall'indirizzo, oppure con l'istruzione LOAD"nome",8,1, nel qual caso il caricamento avviene a partire dall'indirizzo specificato dal file stesso. Un secondo tipo di file, usato essenzialmente per la registrazione dei dati, è il sequenziale, indicato dalla sigla SEQ. Un file sequenziale consiste semplicemente in una sequenza di dati di qualsiasi tipo: numerici, alfanumerici o altro. Un terzo tipo di file utilizzato per immagazzinare dati, più complesso da gestire, ma più potente del sequenziale, è il modo relativo (REL). Anche i file relativi possono contenere dati di qualsiasi tipo, ma a differenza dei sequenziali è possibile accedere direttamente al dato desiderato semplicemente indicandone la posizione, mentre con i sequenziali è necessario caricare l'intera sequenza. Il vantaggio è considerevole quando si vuole gestire una grande quantità di dati, e ovviamente lo si paga con una maggiore difficoltà di programmazione, comunque non eccessiva. L'ultimo tipo di file è detto USR (USeR o utente). Questi tipi di file consentono l'accesso diretto al disco, ancora più diretto dei file relativi, infatti in questo caso per caricare un dato si deve addirittura specificare la traccia e il settore in cui si trova. Questo sistema è poco usato, poiché non è molto pratico, tuttavia è utile esserne a conoscenza. Vediamo ora un dato importantissimo: la velocità di funzionamento. Purtroppo il 1541-II ha conservato l'incredibile lentezza di funzionamento del 1541, e senza un adeguato velocizzatore è davvero insopportabile. Un file PRG da 202 blocchi richiede ben due minuti e venti secondi per essere registrato e due minuti e dieci secondi per essere caricato. Per cancellare lo stesso file sono necessari venticinque secondi, mentre per formattare un disco ci vuole un minuto e quindici secondi. Il tempo di accesso diretto non è certamente brillante: uno spostamento dalla traccia 1 alla traccia 35

richiede oltre un secondo. Con il diffusissimo velocizzatore Speeddos, compatibile con il 99.9% dei programmi e affidabilissimo, il tempo di caricamento di un file PRG da 202 blocchi si riduce a meno di venti secondi: questo dimostra che le caratteristiche meccaniche ed elettriche del drive sono buone.

Una caratteristica negativa del 1541-II rispetto agli altri drive Commodore è la rumorosità di funzionamento: il passaggio della testina da una traccia all'altra è accettabile, ma il rumore prodotto dal motorino che fa ruotare il dischetto è davvero eccessivo.

Il manuale è esattamente identico al manuale del vecchio 1541, a parte le modifiche ai disegni dovute alla differente forma dell'apparecchio. Questo manuale è senz'altro esauriente per chi desidera semplicemente utilizzare il drive per registrare i propri programmi, ma non è certamente l'ideale per imparare a trattare i file relativi o USR. Terminiamo però con una nota positiva, dovuta all'aggiunta al 1541-II di un utilissimo commutatore che permette di selezionare il numero di device del drive, che può variare da 8 a 11 senza effettuare tagli o saldature di ponticelli come nel vecchio 1541.

Gianni Arioli

Prosegue l'eserciziario di matematica con lo studio delle funzioni trascendenti. Verranno svolti, con l'aiuto del computer, tre esercizi completi sulle funzioni trigonometriche, logaritmiche ed esponenziali.

Funzioni trascendenti

Continua la serie di programmi didattici presentati da RadioElettronica & Computer orientati a facilitare l'apprendimento di alcuni argomenti fra i più ostici.

In questo numero affrontiamo e concludiamo il discorso relativo allo studio di funzione, presentando tre esercizi completi volti ad esaurire il campo delle funzioni che si possono presentare nei vostri esercizi.

Le funzioni prese in considerazione sono le trigonometriche, le logaritmiche e le esponenziali, che sono le funzioni di maggiore difficoltà affrontate dagli studenti delle medie. Altre funzioni possono essere ottenute come combinazioni di questi tre tipi fondamentali, come per esempio le funzioni iperboliche, crediamo quindi di fornire un panorama sufficientementeampio degli esercizidi analisi matematica degli studi medi speriori.

Tenete inoltre presente che lo studio di funzione non è certo un esercizio fine a se stesso, ma interviene molto più di quanto si pensi nella vita di tutti i giorni.

Molto spesso ci si trova di fronte al problema di massimizzare una funzione o di minimizzarne un'altra, oppure si vuole sapere quando una variabile raggiunge uno stato stabile e via dicendo.

In ogni caso, ogni qual volta occorra uno studio su una funzione di qualsiasi natura, sono necessari diversi passaggi obbligati per arrivare al risultato.

Molto spesso il disordine e la disorganizzazione con cui si affrontano le varie fasi del problema sono causa di errore e perciò vorremmo fornirvi una traccia da seguire per facilitarvi il compito.

Figura 1. Funzione trigonometrica.

Come procedere

a. Determinazione dell'insieme T di definizione (è ovvio osservare quanto sia importante questa fase: una scorretta determinazione

Figura 2. Funzione logaritmica.

di tale intervallo compromette spesso tutto lo studio).

b. Controllo di eventuali simmetrie e periodicità. Se la funzione è pari o dispari basterà studiarla per x>=0; se è periodica di periodo P, basterà studiarla in un qualsiasi intervallo di pari ampiezza.

c. Calcolo dei limiti destri e sinistri nei punti di frontiera dell'insieme T e dei limiti all'infinito (se T e illimitato). Questo calcolo permetterà di scoprire l'esistenza di eventuali asintoti.

d. Determinazione degli eventuali punti di incontro con gli assi (è opportuno cercare i punti di incontro con l'asse x solo se l'equazione che li determina è risolubile elementarmente).

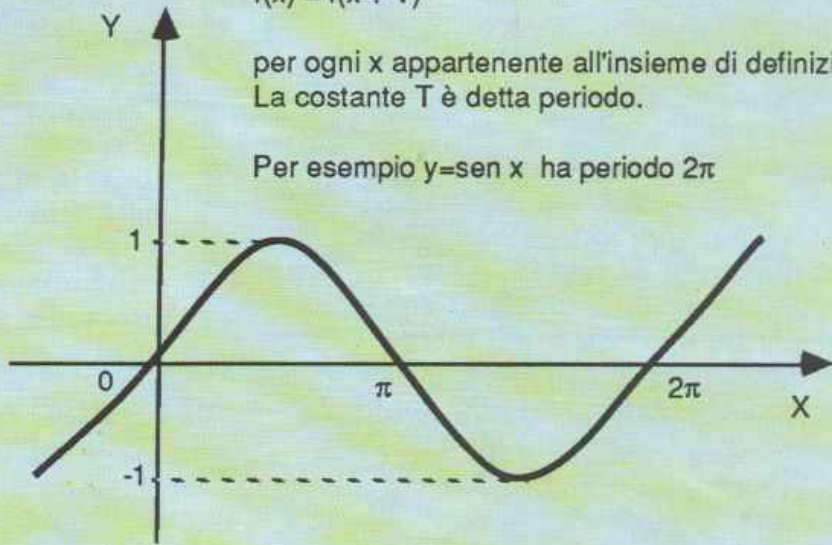
e. Calcolo delle derivate prima e seconda.

f. Studio della derivata prima con preventiva determinazione del suo insieme T' di definizione. Nei punti corrispondenti ai valori di x per cui risulta f'(x)=0 la tangente al grafico è parallela all'asse x.

Dal segno della derivata si ricavano informazioni sulla crescenza e decrescenza della funzione, e di riflesso, sulla locazione di eventuali punti di massimo o di minimo.

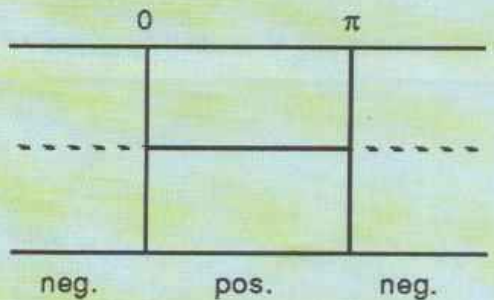
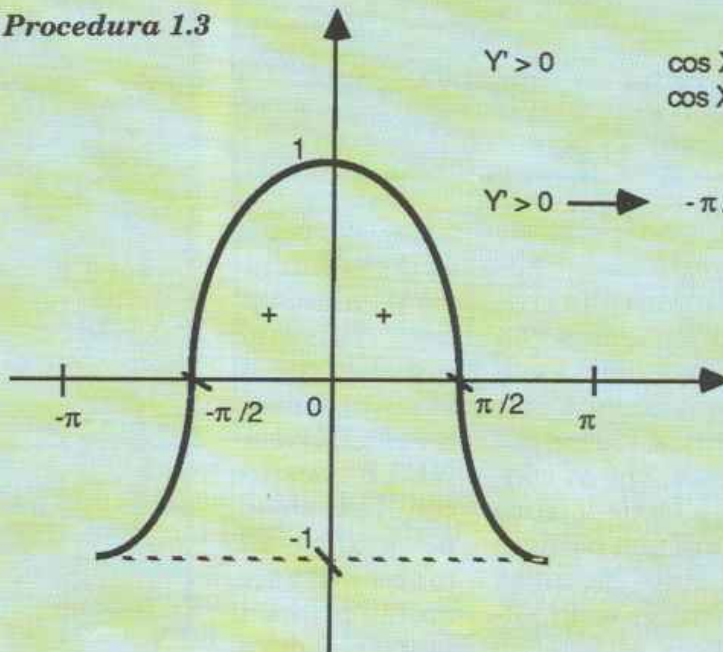
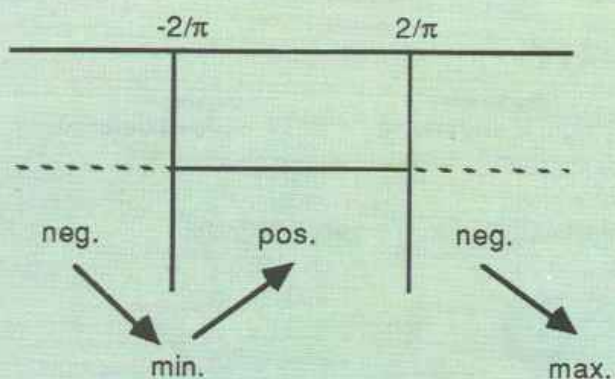
Nel caso lo studio del segno sia particolarmente difficile, vale la seguente regola: " Se f'(x1)=0 e f''(x1)<>0, la funzione ammette nel punto di ascissa x1 un massimo se f''(x1)<0 ed un minimo se f''(x1)>0. "

È necessario anche calcolare i limiti negli estremi di T' in quanto ciò permette l'individuazione di punti angolosi o di cuspide o flessi a tangente orizzontale o, più semplicemente, l'inclinazione della tangente quando il grafico si avvicina al punto critico.

g. Studio della derivata seconda. Serve per sapere la concavità della curva e per valutare i punti di flesso. Se la derivata seconda non risulta studiabile si può ricorrere alla seguente regola: " se f''(x1)=0 e f'''(x1)<>0, nel punto di ascissa xc'è un punto di flesso.

h. Determinazione delle ordinate di tutti i punti che si ritengono necessari per un corretto disegno del grafico.

i. Ricerca di eventuali asintoti obliqui seguendo il metodo già esposto nel seguente numero.

Questo è un buon metodo per affrontare lo studio di funzione in quanto consente di verificare i risultati a mano a mano che si ottengono, con quelli ottenuti in prece-

denza, in modo da rendersi conto degli eventuali errori.

Nella cassetta allegata alla rivista troverete un programma con alcuni esercizi: spesso nello svolgimento troverete richiami a procedure non presenti a video, ma presenti sulla rivista, che vi consenteranno una più facile lettura e vi permetteranno di non perdere la concentrazione sulla globalità dell'esercizio. Inoltre vi sono anche preziosi consigli all'interno del programma su come impostare determinati problemi e su come risparmiare tempo e fatica.

Programma

Sulla cassetta allegata alla rivista troverete, come sempre, il programma relativo alla puntata corrente dell'eserciziario di matematica, registrato con il nome di studio di ST.FUNZIONE2.È scritto con l'espansione grafica Simons Basic, quindi prima di lanciarlo dovete attivare questa espansione,registrata sulla cassetta subito dopo la routine dsave.

Per l'attivazione è sufficiente caricare l'espansione con Shift Run/Stop e quando sullo schermo appare la videata di power up caricare ST. FUNZIONE 2. Potrete trasferire il programma su disco semplicemente con l'istruzione Basic Save, infatti non è necessario ricorrere alla routine dsave quando il programma è, come in questo caso, di dimensioni inferiori all'area destinata al Basic. Una volta lanciato l'eserciziario premete il menù con i 3 esercizi disponibili, relativi a tre tipi di funzioni affrontate in questa puntata:

1. Funzione trigonometrica.
2. Funzione logaritmica.
3. Funzione esponenziale.

4. Fine lavoro.

La selezione avviene premendo il tasto corrispondente al numero riportato sul menù.

La quarta opzione resetta il Computer con un classico SYS 64738, che cancella il programma in memoria, ma lascia attivato il Simons Basic.

Selezionate con attenzione la funzione perché non è prevista l'opzione per interrompere l'esecuzione e tornare al menù. Ogni esercizio è completamente indipendente dagli altri anche se possono essere rilevate numerose analogie nelle tecniche risolutive.

Lo svolgimento è suddiviso in pagine che vanno obbligatoriamente consultate sequenzialmente premendo il tasto Return per passare alla pagina successiva. Come ultima pagina viene presentato il grafico della funzione, con tanto di scalatura.

Per ragioni di praticità e visibilità, non sapendo a priori dove verrà disegnata la funzione, le indicazioni sulle coordinate non sono riportate esplicitamente nel grafico, ma possono essere calcolate in base alle indicazioni della scala, presenti a video in ogni esercizio.

Anche lo svolgimento non è completo a video perché il numero eccessivo di pagine avrebbe appesantito la consultazione.

Si è deciso, come nella scorsa puntata, di utilizzare lo spazio sulla rivista per puntualizzare alcune procedure particolarmente importanti, inserendo i riferimenti alle figure dell'articolo.

Nel prossimo numero parleremo di integrali e vedrete che le sorprese non mancheranno.

Alberto Palazzo

Questo programma sintetizza le migliori caratteristiche dei più famosi sprite editor, aggiungendovi funzioni studiate per la realizzazione di animazioni e la creazione di giochi. È indispensabile il disk drive.

Animation Construction Kit

Le enormi potenzialità grafiche e di animazione di questo straordinario home sarebbero inutilizzabili se la definizione degli sprite dovesse essere introdotta in forma numerica.

Si sono succedute innumerevoli implementazioni, in Basic, in linguaggio macchina e miste, alcune più ricche di altre, ma sempre limitate alla fase di disegno, mentre per l'animazione ogni programmatore finora ha sempre dovuto arrangiarsi con le proprie routine.

Animation Construction Kit è stato creato per colmare questo vuoto e per comporre in un unico programma le migliori caratteristiche degli sprite editor più famosi e le funzioni più efficaci per sviluppare animazioni e videogiochi.

Come funziona

ACK è scritto in linguaggio macchina e si colloca da $0801 a $1D10 e si attiva con un Run.

Sono indispensabili il disk drive e il joystick in porta 2. In cima allo schermo sono visualizzati 8 sprite.

Appena dato il Run tutti e 8 gli sprite utilizzano i dati dello sprite numero 16. Il primo sprite, il numero 0, è lo sprite attivo.

Lo sprite attivo è sempre evidenziato dal lampeggio del numero di definizione.

Ogni sprite è visualizzato in un colore diverso che è anche il colore di default del computer per i suoi 8 sprite.

Dopo aver premuto Shift-CLR/Home la griglia di disegno diventerà completamente blu e il blocco di dati sprite numero 16 verrà azzerato.

Il quadratino lampeggiante in alto a sinistra è il cursore.

Potete utilizzare il joystick per spostare il cursore sulla griglia di disegno.

Premete il pulsante di Fire per plottare un punto sulla griglia, vedrete il punto acceso apparire in ognuno degli 8 sprite in alto sullo schermo.

Potete disegnare linee continue tenendo premuto il pulsante di Fire e spostando contemporaneamente la leva del joystick.

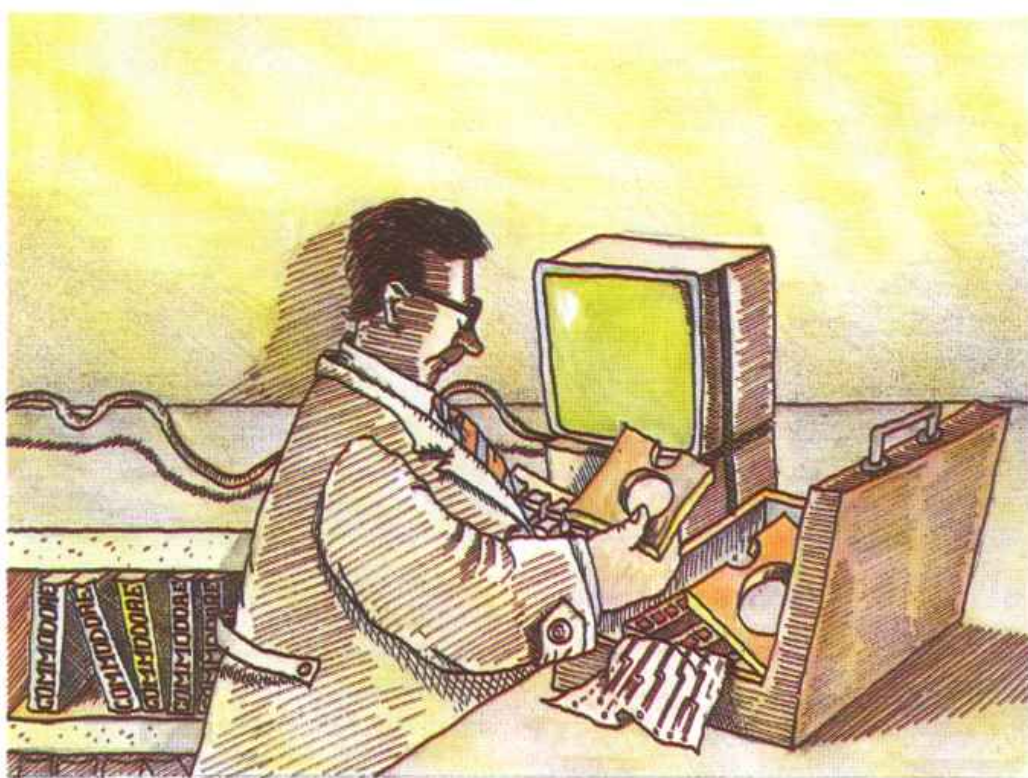
Per cancellare punti o linee basta premere il tasto 1 (penna in modo cancellazione) e tracciare con il joystick.

Per riportare la penna in modo normale basta premere il tasto 3.

Premendo Shift-F lo sprite viene invertito orizzontalmnete, mentre il solo tasto F consente di capovolgere lo sprite.

Premendo i tasti di movimento cursore CRSR Left e Right lo sprite sulla griglia viene scrollato un pixel alla volta nella direzione prescelta.

Analogamente avviene con i ta-

sti di movimenti verticali CRSR up e down.

Salvataggio

Una volta che avrete disegnato tutti gli sprite di cui avete bisogno, potete salvarli premendo il tasto Shift-S.

Si aprirà una finestra e un prompt vi chiederà di specificare i numeri di pagina iniziale e finale dei dati sprite che volete salvare.

La leva del joystick o i tasti di movimento cursore possono essere utilizzati per selezionare i numeri corretti.

Premete la barra spazio per impostare il numero.

Quindi vi verrà chiesto il nome del file con cui salvarli.

Assicuratevi di non inserire le virgolette nel nome del file.

Se all'ultimo istante decidete di non salvare gli sprite, potete interrompere l'operazione premendo Return alla richiesta del nome del file.

Quando avete salvato il vostro file, potete ricaricarlo in ogni momento premendo la combinazione Shift-L.

Vedrete apparire il prompt "Name/Return=Exit", dovrete inserire il nome del file da caricare, omettendo le virgolette o premere semplicemente Return per annullare.

Se commettete un errore nello scrivere il nome del file o se inserite le virgolette, il drive inizierà a lampeggiare e il programma ritornerà alla fase di disegno nella situazione antecedente alla richiesta di Load.

Se vi siete dimenticati il nome del file, potete visualizzare la directory del dischetto premendo Shift-4.

Premete un tasto qualsiasi per scrollare i file della directory o Return per tornare alla fase di disegno.

Premendo Shift-D si attiva l'opzione per salvare gli statement Data contenenti i valori di definizione degli sprite.

L'impostazione dei numeri di pagina è identica a quella già descritta per il salvataggio.

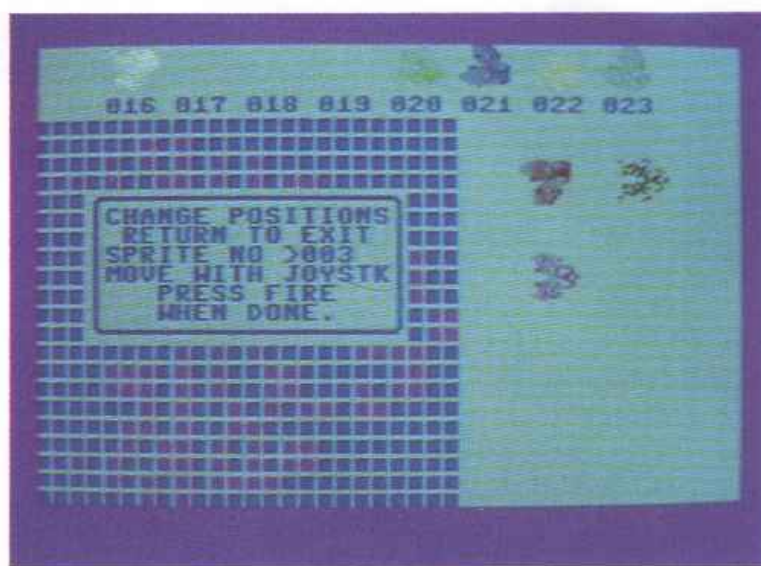
Il file Basic salvato su dischetto conterrà tutte le istruzioni Data numerate a partire dalla linea 1 con incremento 1. Potete renumerarlo direttamente o con una delle tante espansioni o cartridge che hanno l'opzione Renum, per unirlo al vostro programma Basic.

L'animazione

Quando caricate una serie di sprite precedentemente salvata su file, gli 8 sprite visualizzati in alto puntano tutti al blocco di definizione numero 16. Per impostare il blocco di definizione dello sprite attivo, dovete utilizzare i tasti + e -" Se tenete contemporaneamente premuto il tasto Shift, tutti gli 8 puntatori sprite (non solo quello dello sprite attivo), verranno incrementatati o decrementati.

Lo sprite attivo può essere cambiato in ogni momento per mezzo dei tasti funzione, premendo F1 per impostare lo sprite 0, F2 per lo sprite 1 e così via.

Quando state lavorando sugli sprite di un videogioco probabilmente vorrete vedere come viene l'intera sequenza di animazione. Per farlo dovrete utilizzare in modo combinato i tasti funzione e i tasti + e - . Per esempio, fate puntare lo sprite 0 al blocco numero 16, lo sprite 1 al blocco 17, lo sprite 2 al blocco 18 e così via.

Nella creazione di animazioni spesso è necessario disegnare serie di frame o fotogrammi, modificando progressivamente la posizione di ogni frame per dare l'illusione del movimento.

Per salvare un frame nel buffer premete il tasto I.

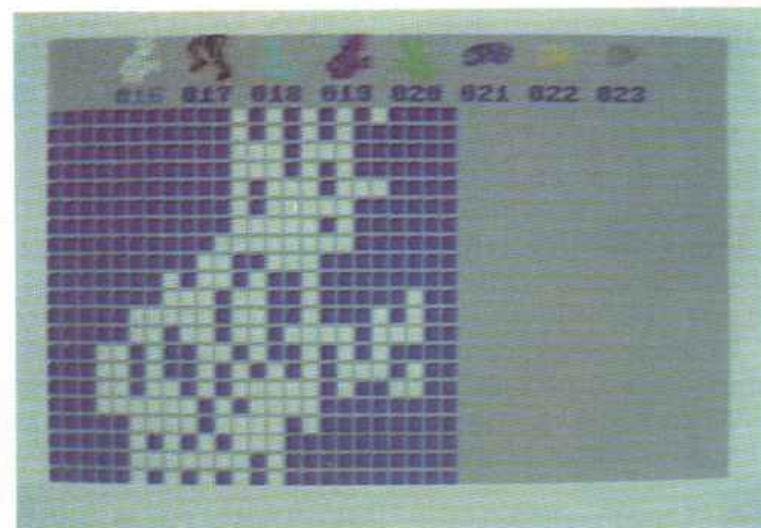
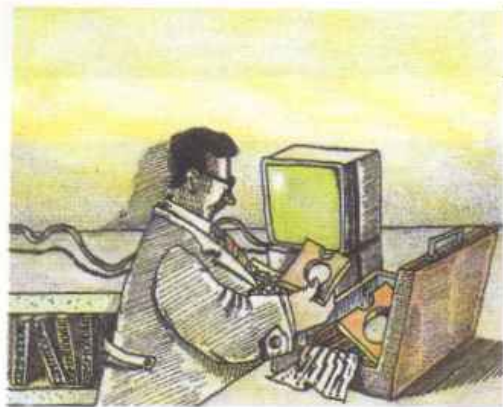
Il buffer tiene un blocco di dati sprite. Per fare pratica impostate uno degli sprite sul blocco 16 con i tasti + e - e poi premete I.

Una finestra lampeggerà con il messaggio "Current data saved". Avanzate il puntatore allo sprite successivo e premete O.

Questo rimpiazzerà i contenuti della pagina corrente con la figura nel buffer.

Altri comandi

Premete il tasto V per modificare la velocità del cursore. Si aprirà una finestra che vi chiederà di scegliere la velocità da un massi-

mo di 1 a un minimo di 9. La velocità del cursore influenza la velocità di disegno e anche la velocità alla quale vengono cambiati i blocchi di definizione degli sprite, con i tasti + e -.

Questa caratteristica vi consente di vedere l'animazione a diverse velocità.

Per tornare a velocità normale, impostate il valore a 5, che corrisponde a quella di default.

Gli sprite visualizzati in cima allo schermo possono essere spostati premendo il tasto Z.

Quindi introducete il numero dello sprite che volete spostare.

Come al solito premete Return per chiuderre la finestra e interrompere il comando.

Per confermare la selezione premete la barra spazio o il pulsante di Fire.

Usate la leva del joystick per spostare lo sprite dove volete sullo schermo.

Infine premete Fire per la nuova posizione.

Questa opzione si rivela particolarmente utile quando dovete comporre figure costituite da più sprite accostati o sovrapposti.

Potete dispore gli sprite su due file da quattro o in gruppi quadrati di 4 per volta secondo le vostre esigenze e in tutta libertà.

Premendo X lo sprite attivo verrà espanso orizzontalmente.

Analogamente premendo Y si otterrà lo stesso effetto verticalmente.

Premendo nuovamente gli stessi tasti lo sprite verrà riportato alle dimensioni normali.

Tenendo premuto anche il tasto Shift il risultato verrà applicato a tutti gli 8 sprite. In alcuni la griglia di disegno può dare astidio sullo schermo.

Per eliminarla basta premere il tasto Shift-W. Premendo W la griglia riappare.

Tavola 1. Riepilogo dei comandi

C	Assegnamento del colore.
D	Sposta lo sprite in basso di un pixel.
DEL	Cancella lo sprite corrente.
HOME	Sposta il cursore in alto a sinistra.
I	Memorizza lo sprite corrente nel buffer.
L	Sposta lo sprite a sinistra di un pixel.
M	Entra/esce dal multicolor.
O	Preleva uno sprite dal buffer.
Q	Termina ACK e riporta al Basic.
R	Sposta lo sprite a destra di un pixel.
S	Incrementa di uno il colore dello schermo.
U	Sposta lo sprite un pixel verso l'alto.
F	Entra ed esce dal modo mirror.
F1-F8	Seleziona uno degli otto sprite.
V	Imposta la velocità del cursore.
W	Rimette a posto la griglia di disegno.
X	Espande/riduce lo sprite in senso orizzontale.
Y	Espande/riduce lo sprite in senso verticale.
Z	Posiziona gli sprite sullo schermo.
+	Incrementa il puntatore di sprite.
-	Decrementa il puntatore di sprite.
$	Visualizza la directory.
CLR	Sbianca lo sprite corrente.
SHIFT F	Attiva/disattiva l'inversione verticale.
SHIFT +	Incrementa tutti i puntatori degli sprite.
SHIFT -	Decrementa tutti i puntatori degli sprite.
SHIFT W	Toglie la griglia di disegno dallo schermo.
SHIFT X	Espande/riduce orizzontalmente gli sprite.
SHIFT Y	Espande/riduce verticalmente gli sprite.
SHIFT M	Entra/esce dal multicolor sugli sprite.
SHIFT L	Carica i dati degli sprite.
SHIFT S	Salva i dati sprite in un file binario.
SHIFT D	Trasforma gli sprite in linee Data .
SHIFT INST	Inserisce un blocco di dati vuoto nella posizione corrente.
SHIFT C	Copia alcuni sprite in un'altra posizione.

Il colore

Ci sono due modi per gli sprite: hires e multicolor.

Iniziamo con il modo Hi-Res. Per modificare il colore di uno sprite è necessario premere il tasto C. Si aprirà una finestra che propone:

Change color of
1 - Background
3 - Sprite color
Return - Exit

Premendo il tasto 1 verrete invitati a premere un tasto colore con la solita combinazione CTRL o CMDR.

Naturalmente se impostate il colore di sfondo uguale a quello di linea non si vedrà più nulla.

Accadrà lo stesso quando si scegli il colore di linea degli sprite.

È possibile cambiare il colore dello sfondo premendo il tasto S, che ha lo scopo di far scorrere ciclicamente i vari colori possibili.

Premendo M si entra in modo multicolor.

Istantaneamente tutti gli sprite vengono visualizzati in multicolor, con la risoluzione orizzontale dimezzata.

Premendo il tasto C si entra nella finestra di scelta dei colori:

1 - Background
2 - M.C.0 [01]
3 - Sprite color

4 - M.C.1 [11]
Return - Exit

La modifica del colore di sfondo funziona esattamente come nel modo Hi-Res.

Premendo il tasto 3, il cambio di colore ha effetto solo sullo sprite corrente. Le opzioni 2 e 4 modificano i colori comuni a tutti gli sprite. Infatti nel modo multicolore ogni sprite può avere 4 colori, ma due sono comuni a tutti gli sprite.

Spostamenti di blocchi

Avete già visto come usare il comando di cancellazione Shift-CLR/Home.

Il comando Delete funziona in modo diverso. Infatti premendo il tasto DEL il blocco corrente di dati sprite viene cancellato e il resto dei blocchi sprite renumerate per riempire lo spazio lasciato libero dallo sprite rimosso.

Questa opzione è molto utile quando è necessario rimuovere un particolare sprite da una sequenza di animazione.

Il comando di Insert, che si attiva con Shift-Inst, inserisce un blocco di dati vuoto nel punto del numero corrente. Gli altri sprite sono renumerati e spostati in avanti di 1, mentre lo sprite 255 viene perso.

Il comando Copy, attivato con Shift-C, copia un gruppo contiguo di sprite in un'altra locazione. Si apre una finestra che chiede il numero del blocco iniziale e finale e il numero del blocco in cui posizionare gli sprite.

I numeri possono essere impostati sia con i tasti cursore CRSR (Left e Right incrementano e decrementano di 10, mentre Up e Down di 1), sia con la leva del joystick.

Per confermare la scelta basta premere la barra spaziatrice.

Il tasto Q vi consentirà di abbandonare ACK e tornare al Basic.

Potete rientrare in ACK semplicemente ridando il Run oppure con SYS2106.

Giuseppe Brigatti

ECCO I COMMODORE POINT E I PUNTI VENDITA AUTORIZZATI CHE CAMBIANO IL TUO VECCHIO COMPUTER CON IL PRINCIPE:

Commodore

COMMODORE POINT - **LOMBARDIA** - MILANO: **Al Risparmio** - V.le Monza, 204 ▪ **Braha Alberto** - Via P. Capponi 5 ▪ **E.D.S.** - C.so Porta Ticinese 4 ▪ **E.S.C.** - Via Roggia Scagna 7 ▪ **Faref** - Via A. Volta 21 ▪ **GBC** - Via Cantoni 7 - Via Petrella 6 ▪ **Giglioni Laura** - Via G. Pacini 4 ▪ **Giglioni** - V.le L. Sturzo 45 ▪ **Logitek** - Via Golgi 60 ▪ **Marcucci** - Via F.lli Bronzetti 37 ▪ **Melchioni** - Via P. Colletta 37 ▪ **Messaggerie Musicali** - Galleria del Corso 2 ▪ **Newel** - Via Mac Mahon 75 ▪ **Rivola** - Via Vitruvio 43 ▪ PROV. DI MILANO: **F.lli Galimberti** - Via Nazionale dei Giovi 28/36 - Barlassina ▪ **GBC** - V.le Matteotti 66 - Cinisello Balsamo ▪ **P. G. Ostellari** - Via Milano 300 - Desio ▪ **Casa della Musica** - Via Indipendenza 21 - Cologno Monzese ▪ **Penati** - Via Verdi 28/30 - Corbetta ▪ **EPM** - V.le Italia 12 - Corsico ▪ **Centro Computer Pandolfi** - Via Corridoni 18 - Legnano ▪ **Computeam** - Via Vecellio 41 - Lissone ▪ **Futura** - Via Solferino 31 - Lodi ▪ **M.B.M.** - C.so Roma 112 - Lodi ▪ **L'Amico del Computer** - V.le Lombardia 17 - Melegnano ▪ **Bit 84** - Via Italia 4 - Monza ▪ **I.C.O.** - Via dei Tigli 14 - Opera ▪ BERGAMO: **Comif** - Via Autolinee 10 ▪ **Cordani** - Via dei Caniana 8 ▪ **D.R.B.** - Via Borgo Palazzo 65 ▪ **New Systems** - Via Paglia 36 ▪ PROV. DI BERGAMO: **Bertulezzi G.** - Via Fantoni 48 - Alzano Lombardo ▪ **Computer Team** - Via Verdi 1/B - Carvico ▪ **Ottico Rovetta** - P.zza Garibaldi 6 - Lovere ▪ **A.I.S. International** - Via San Carlo 25 - San Pellegrino Terme ▪ **Sisthema** - Via Roma 45 - Sarnico ▪ BRESCIA: **Computer Center** - Via Cipro 62 ▪ **Informatica 2000** - Via Stazione 16/B ▪ **Vigasio Mario** - Portici Zanardelli 3 ▪ PROV. DI BRESCIA: **Mister Bit** - Via Mazzini 70 - Breno ▪ **Cavalli Pietro** - Via 10 Giornate 14B - Castrezzato ▪ **Vietti Giuseppe** - Via Milano 1/B - Chiari ▪ **Megabyte** - P.zza Duomo 17 - Desenzano del Garda ▪ **Baresi R. & C.** - Via XX Settembre 7 - Ghedi ▪ **Info Cam** - Via Provinciale 3 - Gratacasolo ▪ COMO: **Il Computer** - Via Indipendenza 90 ▪ **2M Elettronica** - Via Sacco 3 ▪ PROV. DI COMO: **Eltrongros** - Via L. da Vinci 54 - Barzanò ▪ **EGA** - Via Mazzini 42 - Cassago Brianza ▪ **Data Found** - Via A. Volta 4 - Erba ▪ **EGA** - Via A. Moro 17 - Galbiate ▪ **Righi Elettronica** - Via G. Leopardi 26 - Olgiate Comasco ▪ **Cima Elettronica** - Via L. da Vinci 7 - Lecco ▪ **Fumagalli** - Via Cairoli 48 - Lecco ▪ CREMONA: **Mondo Computer** - Via Giuseppina 11/B ▪ **Prisma** - Via Buoso da Dovara 8 ▪ **Telco** - P.zza Marconi 2/A ▪ PROV. DI CREMONA: **Elcom** - Via IV Novembre 56/58 - Crema ▪ **Euroelettronica** - Via XX Settembre - Crema ▪ **Kofbaker** - Via Marchi 65/B - Vescovato ▪ MANTOVA: **Tubaldo E. & C.** - Galleria Fermi 7 ▪ **32 Bit** - Via C. Battisti 14 ▪ **Elettronica di Basso** - V.le Risorgimento 69 ▪ PAVIA: **Poliware** - C.so C. Alberto 76 ▪ PROV. DI PAVIA: **Logica Informatica** - V.le Monte Grappa 32 - Vigevano ▪ **M. Visentin** - C.so V. Emanuele 76 – Vigevano ▪ PIACENZA: **Computer Line** - Via G. Carducci 4 ▪ **Delta Computer** - Via Martiri della Resistenza 15/4 ▪ **Sover** - Via IV Novembre 60 ▪ SONDRIO: **Cipolla Mauro** - Via Tremogge 25 ▪ PROV. DI SONDRIO: **Fotonova** - San Pietro di Berbenno ▪ VARESE: **Dimeco Sistemi** - Via Garibaldi ▪ **Il Centro Elettronico** - Via Morazzone 2 ▪ **Supergames** - Via Carrobbio 13 ▪ PROV. DI VARESE: **Busto Bit** - Via Gavinana 17 - Busto Arsizio ▪ **Crespi G. & C.** - V.le Lombardia 59 - Castellanza ▪ **Computer Shop** - Via A. Da Brescia 2 - Gallarate ▪ **Grandi Magazzini Bossi** - Via Clerici 196 - Gerenzano ▪ **J.A.C.** - Via Matteotti 38 - Sesto Calende ▪ **PIEMONTE** - ALESSANDRIA: **Bit Micro** - Via Mazzini 102 ▪ **West Records** - C.so Roma 85 ▪ PROV. DI ALESSANDRIA: **S.G.E. Elettronica** - Via Bandello 19 - Tortona ▪ ASTI: **Record** - C.so Alfieri 166/3 ▪ CUNEO: **Rossi Computers** - C.so Nizza 42 ▪ PROV. DI CUNEO: **Punto Bit** - C.so Langhe 26/C - Alba ▪ **SDI** - Via V. Emanuele 250 - Bra ▪ **Aschieri G. Franco** - C.so E. Filiberto 6 - Fossano ▪ NOVARA: **Elcom** - C.so Mazzini 11 ▪ **Programma 3** - V.le Buonarroti 8 ▪ **Punto Video** - C.so Risorgimento 39/I ▪ PROV. DI NOVARA: **Mirco Polacco & C.** - Via Monte Zeda 4 - Arona ▪ **All Computer** - C.so Garibaldi 106 - Borgomanero ▪ **Micrologic** - Via Giovanni XXIII 2 - Domodossola ▪ **Elliott Computer** - Via Don Minzoni 32 - Intra ▪ TORINO: **Aba Elettronica** - Via C. Fossati 5/P ▪ **Alex Computer** - C.so Francia 333/4 ▪ **Computing New** - Via M. Polo 40/E ▪ **De Bug** - C.so V. Emanuele II 22 ▪ **Desme Universal** - Via San Secondo 95 ▪ **F.D.S.** - Via Borgaro 86/D ▪ **Computer Home** - Via San Donato 46/D ▪ **Informatica Italia** - C.so Re Umberto 129 ▪ **MT Informatica** - C.so G. Cesare 58 ▪ **New Business Computer** - Via Nizza 45/F ▪ **Radio TV Mirafiori** - C.so Unione Sovietica 381 ▪ **SMT Elettronica** - Via Bibiana 83/B ▪ PROV. DI TORINO: **Paul e Chico Videosound** - Via V. Emanuele 52 - Chieri ▪ **Bit Informatica** - Via V. Emanuele 154 - Ciriè ▪ **Hi-Fi Club** - C.so Francia 92/C - Collegno ▪ **I.C.S.** - Stradale Torino 73 - Ivrea ▪ **BAS** - C.so Roma 47 - Moncalieri ▪ **Cerutti Mauro** - C.so Torino 234 - Pinerolo ▪ **Eurex** - C.so Indipendenza 5 - Rivarolo C.se ▪ VERCELLI: **Elettrogamma** - C.so Bormida ▪ **Elettronica di Bellamo A. & C.** - Strada Torino 15 ▪ PROV. DI VERCELLI: **C.S.I. Teorema** - Via Losana 9 - Biella ▪ **Fotostudio Trevisan** - Via XXV Aprile 24/B - Cossato ▪ **Studio Fotografico Imarisio** - P.zza Martiri Libertà 7 - Trino.

PUNTI VENDITA AUTORIZZATI - **LIGURIA** - GENOVA: **Centro Elettronica** - Via Chiaravagna 10R ▪ **Commerciale Sottoripa** - Via Sottoripa 115R ▪ **Fotomondial** - Via del Campo 35R ▪ **La Nascente** - Via San Luca 26/28R ▪ **Play Time** - Via Gramsci 3/5/7R ▪ **Rappel** - Via Borgoratti 23/I/R ▪ IMPERIA: **Castellino** - Via Nazionale 253 - Via Belgrano 44 ▪ PROV. DI IMPERIA: **Castellino** - Via Roma 68 - Sanremo ▪ **Centro Hi-Fi Video** - Via della Repubblica 38 - Sanremo ▪ **Castellino** - Via Genova 48 - Ventimiglia ▪ LA SPEZIA: **I.L. Elettronica** - Via Lunigiana 618 - Via V. Veneto 123 ▪ PROV. DI LA SPEZIA: **New Soft** - Via Canaletto 5 - Castelnuovo Magra ▪ **Ferrari e Lanzoni** - Via Genova 33/35 - Ceparana ▪ **I.L. Elettronica** - Via Aurelia 299 - Fornola di Vezzano ▪ SAVONA: **Castellino** - C.so Tardy e Benech 101 ▪ **LOMBARDIA** - PROV. DI MANTOVA: **Foto Anna** - Via Matteotti 51 - Poggiorusco ▪ **VENETO** - BELLUNO: **Up to Date** - Via V. Veneto 43 ▪ PADOVA: **Bit Shop** - Via Cairoli ▪ **Computer Point** - Via Roma 63 ▪ **G. Franco Marcato** - Via Madonna della Salute 51/53 ▪ PROV. DI PADOVA: **Compumania** - Riviera Tiso 37 - Camposanpiero ▪ PROV. DI TREVISO: **Ires** - Via Dante 1 - Cessalto ▪ **De Marin Computers** - Via Matteotti 142 - Conegliano ▪ **Sidestreet** - Via S. D'Acquisto 8 - Montebelluna ▪ VENEZIA: **Caputo Ruggero** - P.zza San Marco 5193 ▪ PROV. DI VENEZIA: **Ghegin** - Via Miranese 283 - Chirignago ▪ **Guerra E. & C.** - Via Bissuola 20/A - Mestre ▪ **Paccinotti** - Via Caneve 94 - Mestre ▪ VERONA: **Cartolibreria Fiscale** - Via del Pontiere 24 ▪ **Casa della Radio** - Via Cairoli 10 ▪ **Personal Ware** - Vicolo Volto San Luca ▪ PROV. DI VERONA: **Castagnetti** - Via Strà 19 - Caldiero ▪ **Bussola Luciano** - Via Brennero 54 - Domegliara ▪ PROV. DI VICENZA: **Marangoni Giuseppe** - Via Marconi 8 - Schio ▪ **FRIULI VENEZIA GIULIA** - TRIESTE: **Avanzo Giacomo** - P.zza Cavana 7 - C.so Italia 17 ▪ **Computer Shop** - Via P. Reti 6 ▪ **Universal Tecnica** - C.so Saba 18 ▪ **EMILIA ROMAGNA** - BOLOGNA: **Computer Facile** - Via Don Minzoni 4/B ▪ **Minnella Alta Fedeltà** - Via Mazzini 146/2 ▪ **Morini & Federici** - Via Marconi 28/C ▪ PROV. DI BOLOGNA: **S.P.E. Informatica** - Via di Mezzo Ponente 383/A - Crevalcore ▪ **Archimede Sistemi** - Via Emilia 124 - San Lazzaro di Savena ▪ FERRARA: **Soft-Gallery** - Via Mortara 30 ▪ FORLÌ: **Computer Video Center** - Via Campo di Marte 122 ▪ **Ellegi Computer** - C.so Garibaldi 87 ▪ PROV. DI FORLÌ: **Top Bit** - Via Veneto 12 - Forlimpopoli ▪ **Computer House** - V.le Tripoli 193/d - Rimini ▪ MODENA: **Centro Calcolo** - Via Muzzioli 18 ▪ **Orsa Maggiore** - P.zza Matteotti 20 ▪ PROV. DI MODENA: **Centro Calcolo** - Via Muratori 3 - Carpi ▪ **Elettronica Ferretti** - Via Cialdini 41 - Sassuolo ▪ **Bertoni B. Machines** - C.so Italia 28 - Vignola ▪ PROV. DI RAVENNA: **Argnani** - P.zza Libertà 5/A - Faenza ▪ REGGIO EMILIA: **Computer Line** - Via San Rocco 10/C ▪ **Pool Shop** - Via Emilia S. Stefano 9/C ▪ PROV. DI REGGIO EMILIA: **Macchioni Armando** - Via Statale 467,27 - Casalgrande ▪ **TOSCANA** - AREZZO: **Delta System** - Via Piave 13 ▪ FIRENZE: **Atema** - Via B. Marcello 1/A ▪ **Caff** - Via Allori 52 ▪ **Elettronica Cento Stelle** - Via Cento Stelle 5/A ▪ **New Computer Service** - Via degli Alfani 2/R ▪ **TIT** - Via Bronzini 36 ▪ PROV. DI FIRENZE: **War Games** - Via R. Sanzio 126 - Empoli ▪ **New EVM** - Via degli Innocenti 2 - Figline Valdarno ▪ **Atema** - Via Pisana 407 - Scandicci ▪ GROSSETO: **Tutto Computer** - Via Gramsci 2/A ▪ LIVORNO: **Eta Beta** - Via San Francesco 30 ▪ **Futura 2** - Via Cambini 19 ▪ PROV. DI LIVORNO: **Bonanni Elettronica** - Via Rosmini 3/A - Cecina ▪ **Radio Mechi** - C.so Matteotti 124 - Cecina ▪ **Tesi Adriano** - Via Carducci 4 - Piombino ▪ **Tomi Gusmano** - Via Petrarca 109 - Piombino ▪ PROV. DI LUCCA: **Il Computer** - V.le Colombo 216 - Lido di Camaiore ▪ **Pellegrini 1** - Via Oberdan 65 - Pietrasanta ▪ **Logica** - Via A. Fratti 165 - Viareggio ▪ PROV. DI MASSA: **Giorgi G. & C.** - Via San Leonardo 350 - Marina di Massa ▪ PISA: **C.H.S.** - Via C. Cattaneo 90/92 ▪ **Electronic Service** - Via della Vecchia Tramvia 10 ▪ **IT-LAB** - Via Marche 8/A/B ▪ **Tony Hi-Fi** - Via Carducci ▪ SIENA: **Video Movie** - Via Garibaldi 17 ▪ PROV. DI SIENA: **Bifolchi Giordano** - Via di Gracciano nel Corso 111 - Montepulciano ▪ **UMBRIA** - PERUGIA: **Migliorati Piero** - Via S. Ercolano 3/10 ▪ PROV. DI PERUGIA: **Computer Studios** - Via IV Novembre 18/A - Bastia Umbra ▪ **Tonzani Orietta** - Via G. Di Vittorio 13/A/B/C - Ellera ▪ **Marinelli Elettronica** - Via Mazzini 104 - Foligno ▪ TERNI: **Ramozzi Rossana** - Via Porta S. Angelo 23 ▪ **MARCHE: PROV. DI ANCONA: Bit e Video** - C.so Matteotti 28 - Jesi ▪ PROV. DI ASCOLI PICENO: **Zerouno Computer** - Via Voltatorni - San Benedetto del Tronto ▪ MACERATA: **Giannobi Muzio** - C.so Cavour 93 ▪ **LAZIO** - PROV. DI LATINA: **Marcheggiani Luigi** - Via G. Verdi 64 - Aprilia ▪ ROMA: **Big Byte** - Via V.G. De Vecchi Pieralice 35 ▪ **Compushop** - Via Nomentana 265 ▪ **Computer Friend** - Via A. Romano 3 ▪ **Computron** - L.go Forano 7 ▪ **Data Power** - Via di Fontana Candida 2/C ▪ **Due Emme Elettronica** - Via Britannia 17 ▪ **Egis Computer** - Via Castro dei Volsci 42 ▪ **Elettromarket** - Via C. Balbo 1 ▪ **La Placa Vincenzo** - Via Val Trompia 12/18 ▪ **Leonardo** - Via Chopin 29 - Via P. Castaldi 7 ▪ **Metro Import** - Via Donatello 37 ▪ **Musical Cherubini** - Via Tiburtina 360 ▪ **Pix Computer** - Via F. D'Ovidio 6/C ▪ **Romana Componenti Elettronici** - P.zza dei Gerani 40/41 ▪ **R.P.M.** - Via Giulia 142 ▪ PROV. DI ROMA: **Delta Bit** - Via G. Verdi 26 - Albano Laziale ▪ **Computer Time** - Via Col di Lana 11/15 - Ciampino ▪ **L'Angolo del Computer** - Via delle Case Nuove 3 - Civitavecchia ▪ **M.R.S.** - Via L. Manara 11 - Frascati ▪ **Marzetti Elettrodomestici** - C.so V. Colonna 11 - Marino ▪ **Bit House** - Via Kennedy 100 - Monterotondo ▪ **Lab. P. D'Alberti** - V.le dei Promontori 148 - Ostia Lido ▪ **Paolini** - Via Paolini 94 - Ostia Lido ▪ **A.V.C.** - Via Empolitana 134 - Tivoli ▪ **Computer Shop** - Via 2 Giugno 34 - Tivoli ▪ VITERBO: **Treac** - Via Palazzina 1 ▪ **ABRUZZO** - PESCARA: **Chip Computer** - Via Milano 77/8 - Via N. Adriatica Nord 386 ▪ **BASILICATA** - MATERA: **G. Gaudiano Electronics** - Via Roma 2 ▪ **PUGLIA** - BARI: **Artel** - Via G. D'Orso 9 ▪ **Computer's Arts** - Via Re David 171 ▪ **Discorama** - C.so Cavour 99 ▪ PROV. DI BARI: **Zingaro e Zagaria** - Via Torino 26/28 - Andria ▪ **Faggella Gianni** - Via Alvisi 4 - Barletta ▪ **Pietrantonio G. & C.** - Via Vavalle 2/A - Conversano ▪ BRINDISI: **Olivieri** - Via Bezzecca 9 ▪ **FOGGIA: Botticelli Guido** - Via Sav. Pollice 2 ▪ **La Torre** - V.le Michelangelo 185 ▪ LECCE: **Bit** - Via 95° Reg. Fanteria 87/89 ▪ TARANTO: **Elettrojolly** - Via De Cesare 13 ▪ **T E A** - Via Regina Elena 101 ▪ **SICILIA** - CATANIA: **A Zeta** - Via Canfora 140 ▪ **C.D.M.P.** - Via Amantea 4 ▪ **Elettronica Delta** - Via Messina 413 B ▪ **Foto Ottica Randazzo** - L.go dei Vespri 21 ▪ **Paratore** - Via Maddem 141 ▪ MESSINA: **Mister Bit** - Via Nazionale 10 ▪ **Office Automation** - Via G. Venezian 75 ▪ PALERMO: **Home Computers** - V.le delle Alpi 50/F ▪ **Randazzo Angelo** - Via R. Settimo 53/55 - Via Lulli 20/40 ▪ RAGUSA: **Medi Informatica** - Via Risorgimento 54 ▪ PROV. DI RAGUSA: **Giannone Computers** - Via Vanella Macallè - Modica ▪ SIRACUSA: **Computer Soft Center** - Via San Simeone 15 ▪ PROV. DI SIRACUSA: **Amore Maria Inserra & C.** - Via Garibaldi 15 - Lentini.

Il MIDI è ormai universalmente accettato come lo standard dell'informatica musicale. Vedremo, in questa serie di articoli, le caratteristiche del linguaggio, le specifiche hardware e le più interessanti applicazioni.

Lo standard MIDI

Nel corso dell'evoluzione degli strumenti musicali e delle relative possibilità interpretative ed espressive della musica che ne è conseguita, a partire dai primi decenni di questo secolo sono state raggiunte tappe significative fra le quali si possono menzionare: l'organo Hammond; l'organo elettronico; il sintetizzatore analogico; il sintetizzatore digitale; il campionatore.

La più significativa svolta è però avvenuta nel 1983 con l'introduzione del MIDI.

Cosa è il MIDI

MIDI è l'acronimo di Musical Instrument Digital Interface. Esso rappresenta uno standard creato dall'industria elettronica musicale per collegare e far comunicare strumenti musicali elettronici quali sintetizzatori, drum machines, sequencers e altre diavolerie che oggi nascono in continuazione grazie all'evoluzione

della componentistica e che permettono di realizzare sistemi dalle prestazioni sempre più sofisticate. Già dall'avvento dei primi sintetizzatori analogici, il musicista sentiva il bisogno di ottenere un accompagnamento automatico o di sincronizzare una Drum machine con un Sequencer ma, a causa della mancanza di uno standard fra i vari costruttori, il gioco era possibile solo fra apparati omogenei della stessa marca; addirittura, a volte, generazioni diverse di apparati non comunicavano tra loro.

Forse il movente che ha spinto i principali nomi dell'industria a collaborare per definire e adottare uno standard è stato l'enorme potenziale di vendita che, in un effetto di sinergia, li avrebbe avvantaggiati tutti insieme. Ciò si è avverato puntualmente e la standardizzazione ha incoraggiato le vendite che, a loro volta, hanno abbassato i costi, consentendo una maggiore diffusione dei prodotti. Un altro grande vantaggio della standardizzazione MIDI è stato quello di poter dare all'utente la possibilità di realizzare un proprio sistema a piacere composto da attrezzature anche di marche diverse. Il mercato dell'offerta si presenta infatti segmentato sia in pochi grossi costruttori che hanno a catalogo una linea di prodotti incredibile, sia in molte ditte specializzate solo in determinati componenti o accessori.

Apparati MIDI

Oggi troviamo in commercio, equipaggiate con il MIDI, numerosi apparati e accessori, per i quali, al fine di conoscerli meglio, tentiamo la seguente classificazione per famiglie:

A) Generatori di suono *(sintetizzatori polifonici).* Se sono dotati di tastiera vengono chiamati synth o più semplicemente tastiere. I synth più comuni sono anche disponibili in versione rack, di 19" di larghezza standard, privi di tastiera. In questo caso si chiamano expander. I synth si possono suddividere in funzione del tipo di generazione del suono nelle seguenti categorie: 1) Analogici a controllo digitale; 2) Digitali FM; 3) Digitali a distorsione di fase; 4) Campionatori; 4) Altri. Oltre ai synth vanno menzionati i piani digitali, le chitarre o gli adattatori per chitarre MIDI, eccetera. Non si vuole qui entrare nel merito di ciascun tipo di generazione, quindi, prescindendo dalle peculiarità di ciascun strumento, possiamo semplificare dicendo che i synth presentano in generale al musicista le seguenti funzioni:

- Scelta dei timbri entro un certo numero (32, 64, ecc.) presettati in ROM e altri da creare a cura dell'utente in RAM. Attenzione che a volte i timbri vengono chiamati voci e ciò può confondere per il fatto che anche i generatori del synth si chiamano voci; forse conviene utilizzare il modo di chiamare i timbri adottato nel MIDI e cioè Programs).
- Variazioni dei timbri, editing e creazione di nuovi timbri.
- Variazione dei parametri della tastiera ai fini dell'esecuzione (split, pitch bend, modulation, trasposizione, vibrato, chorus, eccetera).
- Predisposizioni del funzionamento MIDI.

B) Drum machine *(batterie elettroniche)*

C) Accessori. Non generano suono, ma lo modificano oppure controllano quelli della categoria A): riverberi, pedali, midi box, centralini per luci, mixers.

D) Sequencer. Mettiamo qui quelli dedicati.

E) Computers. I computer si collegano al mondo MIDI tramite apposite interfacce (come vedremo più avanti) e possono essere utilizzati per controllare uno o più strumenti collegati. Il software MIDI è generalmente di tre tipi:

• *Sequencer.* Di impiego generalizzato si adattano a tutti gli strumenti della sopracitata categoria A.

• *File ed editor di timbri.* Sono software dedicati a specifiche macchine in quanto lavorano con i codici di sistema esclusivo della macchina.

Lettura dell'Implementation Chart

La MIDI Implementation Chart è una tavola che riassume schematicamente i messaggi MIDI trasmessi ricevuti da uno strumento. Collegando il MIDI OUT di uno strumento A al MIDI IN di uno strumento B, occorre far riferimento alla parte TRASMITTED per conoscere i messaggi iniziabili dall'apparecchio A, mentre è necessario basarsi sulla parte RECOGNIZED per sapere quali siano i messaggi che l'apparecchio B può ricevere. La MIDI Implementetion Chart è divisa in 4 colonne. La prima colonna definisce le funzioni MIDI; la seconda ci informa circa la trasmissione delle funzioni; la terza circa la ricezione; la quarta è riservata alle annotazioni. Vengono utilizzati i simboli O e X, che stanno rispettivamente per SI e NO, cioè esistenza o meno della funzione. L'asterisco *, ci rimanda alle annotazioni presenti sul fondo della tabella.

Figura 1A. Rhythm machine Model TR-505 MIDI Implementation Chart

Function		Transmitted	Recognized	Remarks
Basic Channel	Default Changed	1-6 1-6	1-6 1-6	Memorized
Mode	Default Messages Altered	Mode 3 **************	Mode 1 OMNI ON/OFF	
Note Number	 True voice	25-99 *1 ***************	25-99 *1	assigns to each rhythm voice
Velocity	Note ON Note OFF	O 9n v=48-127 X 9n v=0	O 9b v=1-127 X	n=Inst CH *2 b=Basic CH
After Touch	Key's Ch's	X X	X X	
Pitch Bender		X	X	
Control Change		X	X	
Prog Change	 True #	X *********	X	
System Exclusive		X	X	
System Common	Song Pos Song Sel Tune	O O X	O SYNC mode= MIDI O SYNC mode=MIDI X	 0-5
Sistem Real Time	Clock Commands	O O	O SYNC mode=MIDI O SYNC mode=MIDI	
Aux Mes-Sages	Local ON/OFF All Notes OFF Active Sense Reset	X X X X	X X X X	
Notes		*1 Transmitted and recognized note numbers can be assigned by pannel operation. *2 Transmit channel # for each instrument can be changed to 1-16 pannel operation		

Mode 1 : OMNI ON. POLY
Mode 3 : OMNI OFF. POLY
Mode 2 : OMNI ON.MONO.
Mode 4 : OMNI OFF. MONO

O: YES
X: NO

Figura 1B. Voice Digital Keyboard MIDI Implementation Chart

Function		Transmitted	Recognized	Remarks
Basic Channel	Default Changed	1-6 1-6	1-6 1-6	Memorized
Mode	Default Messages Altered	Mode 3 POLY, OMNI OFF **********	Mode 1, 3, 4 Mono, Poly, Omny On/Off Mode 2 → Mode 1	Memorized
Note Number	True voice	12 - 108 *****	0 - 127 1 - 108	
Velocity	Note ON Note ON	O X 9n v=0	O v=1-127 X	
After Touch	Key's Ch's	X *	X *	
Pitch Bender		*	* 0- 12 semi	9 bit resolution
Control Change	1 5 7 0-31 6, 38 64 65 64-95 100, 101	* * * O X * * O X	* * * O (0, 2-4, 8-31) ** * * O (66-95) ** (0, 1)	Modulation Portamento Time Volume Ext control Data Entry (MBS, LSB) Hold 1 Portamento SW Pedal Switch RPC (LSB, MSB)
Prog Change	True #	* 0 -127 *********	* 0-127 0-127	
System Exclusive		*	*	
System Common	Song Pos Song Sel True	X X X	X X X	
Sistem Real Time	Clock Commands	X X	X X	
Aux Message	Local ON/OFF All Notes OFF Active Sense Reset	X O (123) X X	O O (123-127) O X	Memorized
Notes		* Can be set to O or X manually, and memorized. ** RPC=Registred parameter control number. RPC # 0 : Pitch bend sensitivity RPC # 1 : Master fine tuning Parameter values are given by Data Entry.		

Mode 1 : OMNI ON. POLY **Mode 2** : OMNI ON.MONO. O: YES
Mode 3 : OMNI OFF. POLY **Mode 4** : OMNI OFF. MONO X: NO

• *Utility*. Consentono funzioni varie fra cui: splits della tastiera in più sezioni, ciascuna delle quali con diverso canale MIDI; traspositori di note; transcodifica di dati MIDI eccetera.

Grazie al MIDI, tutte le macchine possono essere connesse e suonare insieme!

Le tre prese standard degli strumenti Midi.

Cosa si controlla

Abbiamo detto che il MIDI è un sistema che permette a diversi strumenti musicali di comunicare tra loro.

Tramite esso infatti, si può, da uno strumento o da un computer, controllare completamente un altro strumento.

Per fare ciò occorre comunicare allo strumento controllato dal posto di controllo (tastiera, drum machine, computer o altro dispositivo) tutti i dati relativi alle azioni che un musicista compie durante l'esecuzione di un brano. Esse si possono così riassumere:

A) Se si considera un synth:

• *Azioni sui tasti*. Nota premuta (note on), nota rilasciata (note off), pressione dinamica del tasto (velocity), successiva pressione (after touch).

• *Azioni sui controlli dello strumento*. Variazione di un qualunque:

- controllo (continui: pedale del volume, manopole, ruota di modulazione, potenziometri slide oppure interruttori on/off per vibrato, chorus, eccetera).
- cambio di timbri, chiamato program change nei synth, ovvero la scelta dei banchi di memoria relativi ai diversi suoni disponibili sullo strumento (piano, violino, sax, clarino, eccetera).
- ruota del pitch (consente di variare l'intonazione dello strumento e di introdurre effetti di glissato simili alla leva delle chitarre o al trombone a tiro).
- assegnazione dei parametri per il funzionamento MIDI.

B) Se si considera una drum machine:

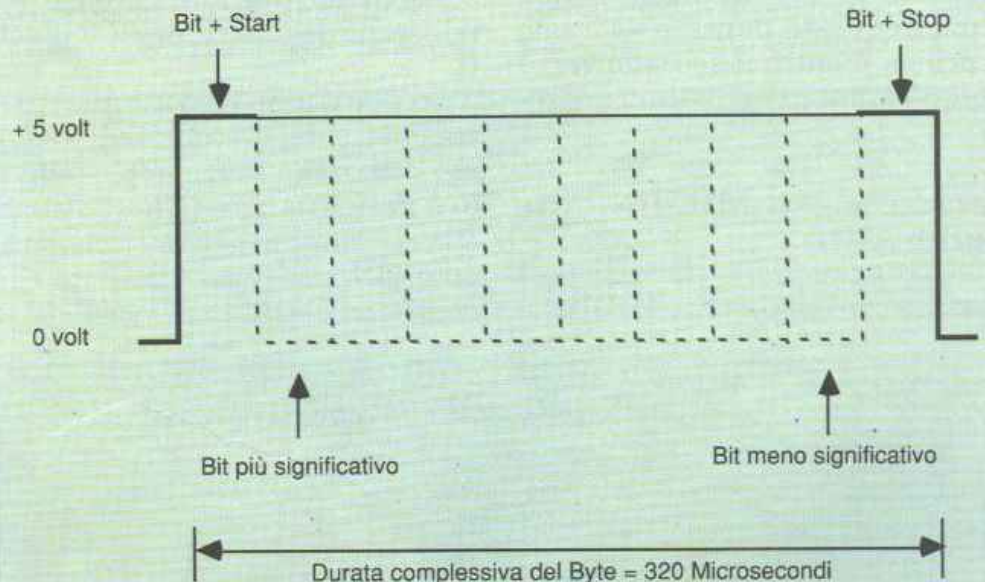
• *Azioni sui tasti* o sui pad dei singoli strumenti. Nota premuta (on), nota rilasciata (off).

• *Azioni su altri controlli:*

- Velocità di esecuzione (timing clock)
- Volume delle varie voci
- Start: inizio delle sequenze
- Stop: interruzione
- Continua: ripresa dal punto di interruzione

• *Assegnazione dei parametri* per il funzionamento MIDI.

C) Se si utilizza un sequencer:

Oltre ad alcuni dei comandi visti per le drum machine, si possono avere i comandi di Song position pointer per selezionare da quale punto deve partire l'ascolto delle sequenze e il Song select per scegliere quale song eseguire fra quelle disponibili.

I byte di stato sono caratterizzati dal Bit più significativo a 1, i byte di dati sono caratterizzati dal Bit più significatrivo a 0

Standard e linguaggio

Il sistema MIDI attuale consiste di un documento chiamato MIDI Specification 1.0 datato 5 agosto 1983.

Per una semplice comprensione del suo contenuto, si possono individuare due temi principali:

• Lo standard MIDI. Individua le caratteristiche e la struttura dell'hardware di interfaccia.

Il grado di adozione dello standard MIDI da parte di uno strumento è chiamato implementazione MIDI.

• Il linguaggio di comunicazione MIDI. È il protocollo di comunicazione fra apparati che utilizzano lo standard MIDI.

Esaminiamo quindi separatamente e in dettaglio questi due temi; in questa puntata vedremo il primo, mentre il secondo verrà approfondito nel prossimo articolo.

Lo standard MIDI: generalità

L'hardware di interfaccia è presente in ogni strumento MIDI ed è pilotato da un microprocessore che, da un lato, dialoga con l'interno dello strumento e, dall'altro, invia all'esterno i dati o interpreta i dati che riceve dall'esterno in linguaggio MIDI.

Gli strumenti dotati di MIDI infatti, comunicano sotto forma di dati tutte le azioni che il musicista sta svolgendo su di essi, oppure, in fase di ricezione, sono in grado di interpretare tutti i comandi esecutivi che gli pervengono da altri strumenti MIDI.

Tutti i moderni strumenti musicali elettronici, dotati di microprocessore, nascono già con l'implementazione MIDI.

L'implementazione MIDI, ovvero il grado di aderenza alle specifiche hardware MIDI (le possibilità di trasmettere o ricevere tutte o solo una parte delle azioni del musicista) di ogni singolo strumento, dipende dal tipo di strumento e, soprattutto, dal suo prezzo.

Possono essere interconnessi anche strumenti con implementazione MIDI diversa, in questo caso avremo delle limitazioni di prestazioni dovute alle carenze intrinseche degli strumenti coinvolti.

Se, per esempio, lo strumento A trasmette la dinamica di tasto, ma lo strumento B è implementato a dinamica fissa, quest'ultimo, alla ricezione di ogni messaggio di nota ON, eseguirà tale nota con dinamica costante, ignorando le eventuali sfumature dinamiche ricevute. Non esiste infatti una implementazione standard per gli strumenti o per gli accessori dotati di MIDI.

Essa viene stabilita sia dalla natura sia in funzione delle prestazioni (quindi del prezzo) dello strumento, che ogni costruttore stabilisce in fase di progetto. Quindi, nell'effettuare la scelta dell'acquisto, occorre fare molta attenzione, in particolare al secondo fattore.

Per orientarsi, esiste un metodo molto preciso: dopo aver stabilito il livello budget disponibile, chiedete al venditore di consultare nel manuale dello strumento la tabella di implementazione MIDI.

La **figura 1** mostra un esempio di tale tabella.

Questo è un documento standard che sintetizza le prestazioni dello strumento.

Dal punto di vista pratico, rispetto agli strumenti elettronici di generazione precedente, l'apparato MIDI è equipaggiato generalmente con tre prese DIN a 5 contatti (di cui 3 sole utilizzate) come illustrato in **figura 2**: MIDI IN è la porta di ricezione attraverso la quale arrivano i dati allo strumento; tali dati sono ripetuti fedelmente all'interno e rispediti sulla porta MIDI THRU per l'eventuale invio a un altro strumento; MIDI OUT è l'uscita in trasmissione dello strumento.

A seconda del tipo di strumento possiamo avere tutte o alcune delle suddette prese (strumenti economici possono omettere il MIDI THRU, le master keyboards possono non essere equipaggiate con il MIDI IN, alcuni tipi di accessori possono duplicare le prese ove necessario).

Figura 4. Collegamento strumenti in serie.

Lo standard MIDI: tecnica

Gli strumenti MIDI, a differenza dei loro predecessori, sono equipaggiati con alcuni circuiti elettronici in più che, pilotati dall'ormai immancabile microprocessore, svolgono le seguenti operazioni basilari:

• Rilevamento di tutte le azioni effettuate dall'esecutore e trasformazione delle stesse in forma numerica per trasmetterle come dati all'esterno dello strumento.

• Trasmissione in forma seriale dei dati visti sopra attraverso l'uscita MIDI.

• Ricezione da un altro strumento di dati MIDI attraverso l'ingresso MIDI IN.

• Decodifica dei dati ricevuti, trasformazione degli stessi in istruzioni che lo strumento esegue immediatamente.

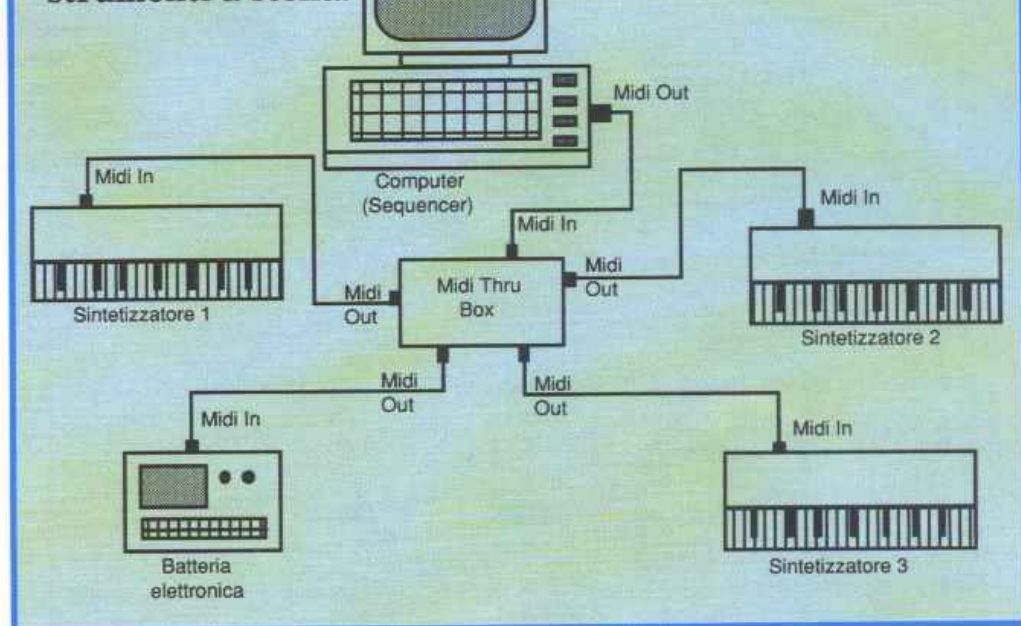
Sono inoltre presenti alcuni controlli (tastini) per predisporre i parametri del funzionamento MIDI dello strumento. Ai fini di questa descrizione del sistema, è interessante occuparci prevalentemente delle interfacce verso l'esterno. Prima di quella MIDI, le interfacce fra strumenti musicali, erano a livello analogico, mentre la MIDI è un sistema completamente digitale.

Il modo più semplice per scambiare dati fra apparati digitali è quello di usare bus paralleli, in quanto, all'interno del computer e, in generale, in apparati a microprocessori come i sintetizzatori, i dati viaggiano su bus a 8 o più fili; tuttavia, ragioni pratiche quali la necessità di usare cavi molto lunghi per collegare fra loro gli strumenti e quindi il rischio di accoppiamenti parassiti e il loro elevato costo, hanno fatto preferire un collegamento di tipo seriale.

Questo consente di usare un semplice cavo, costituito da due fili più uno di massa, con connettori molto economici di lunghezza fino a 15 metri senza problemi.

È stata scelta un'interfaccia seriale asincrona funzionante alla velocità di 31250 Baud molto simile alle interfacce seriali RS 232 C e RS432 usate nei computer.

In un sistema seriale, i dati sono trasmessi su una linea sola, perciò, non essendo possibile trasmettere simultaneamente tutti i bit contenuti in ogni byte di informazione, occorre per ogni byte scomporre in ordine i vari bit e trasmetterli in sequenza uno per

Figura 5. Collegamento strumenti a stella.

uno sull'unica linea a disposizione. Ciò viene fatto, nella maggior parte dei sistemi seriali, iniziando dal bit meno significativo e continuando la sequenza fino al bit più significativo. In pratica sono richiesti 2 fili + 1 di schermatura.

Per ottenere il funzionamento corretto della trasmissione di dati seriali, occorre organizzare il flusso dei dati da trasmettere in modo che il ricevitore sappia il significato dei bit che riceve, li possa ricostruire in forma di byte e interpretarne correttamente il significato. Scartata una forma di sincronizzazione fra trasmettitore e ricevitore per indicare l'inizio di invio di un flusso di dati che avrebbe complicato l'hardware, è stato scelto un sistema asincrono tipico dei sistemi di trasmissione seriali. Usando questo sistema non occorrono fili aggiuntivi lungo la linea di comunicazione per ottenere sincronizzazioni, basta aggiungere qualche bit in più a quelli dei dati da trasmettere per effettuare particolari segnalazioni. Questo compito non riguarda direttamente il progettista hardware, in quanto esistono già cir-

Figura 6.

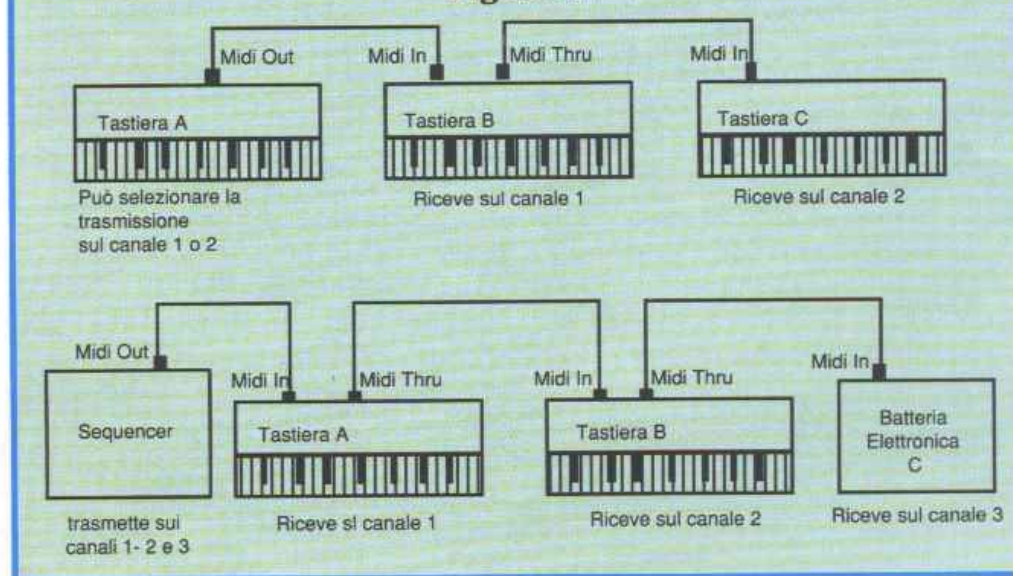



Figura 7. Tabella riassuntiva di tutti i messaggi Midi.

Messaggi di canale

Tipo di messaggio	Valore decimale	Configurazione dei Bit								N. di bytes dati seguenti
Voci										
- Nota rilasciata (Note Off)	Da 128 a 143	1	0	0	0	X	X	X	X	2
- Nota premuta (Note On)	Da 144 a 159	1	0	0	1	X	X	X	X	2
- Pressione di tasto polifonia dopo il tocco (Poly key pressure after touch)	Da 160 a 175	1	0	1	0	X	X	X	X	2
- Variazione di controllo (Control Change)	Da 176 a 191	1	0	1	1	X	X	X	X	2
- Cambio di programma (Program change)	Da 192 a 207	1	1	0	0	X	X	X	X	1
- Pressione di canale dopo il tocco (Channel pressure after touch)	Da 208 a 223	1	1	0	1	X	X	X	X	1
- Variazione ruota intonazione (Pitch wheel change)	Da 224 a 239	1	1	1	0	X	X	X	X	2
Modo										
- Messaggi di modo di canale (Channel mode)	Da 176 a 191	1	0	1	1	X	X	X	X	2

Messaggi di sistema

Tipo di messaggio	Valore decimale	Configurazione dei Bit								N. di bytes dati seguenti
Esclusivi										
- Sistema esclusivo (System exclusive)	240	1	1	1	1	0	0	0	0	Variabile (dipende dallo strumento)
Comuni										
- Puntatore posizione brano (Song position pointer)	242	1	1	1	1	0	0	1	1	2
- Selezione brano (Song select)	243	1	1	1	1	0	0	1	1	1
- Richiesta di accordatura (Tune request)	246	1	1	1	1	0	1	1	0	ø
- Fine sistema esclusivo (End exlusive)	247	1	1	1	1	0	1	1	1	ø
Tempo reale										
- Impulso di sincronismo (Timming clock)	248	1	1	1	1	1	0	0	0	ø
- Partenza da inizio brano (Start)	250	1	1	1	1	1	0	1	0	ø
- Continuo (Continue)	251	1	1	1	1	1	0	1	1	ø
- Arresto (Stop)	252	1	1	1	1	1	1	0	0	ø
- Individuazione attività (Active sensing)	254	1	1	1	1	1	1	1	0	ø
- Reset del sistema (System reset)	255	1	1	1	1	1	1	1	1	ø

Costruttori MIDI

Costruttore	Paese	N.
Sequential Circuits	*USA*	1
Big briar	"	2
Octave plateau	"	3
Moog music	"	4
Bontempi	*Italia*	32
Siel	"	33
Elka	"	47
Kaway	*Giappone*	64
Roland	"	65
Korg	"	66
Yamaha	"	67

Figura 8. Tabella di corrispondenza note musicali/valori decimali dei dati.

OTTAVA	DO♯	DO	RE	RE♯	MI	FA	FA♯	SOL	SOL♯	LA	LA♯	SI♯
—	0	1	2	3	4	5	6	7	8	9	10	11
Ø	12	13	14	15	16	17	18	19	20	21	22	23
1	24	25	26	27	28	29	30	31	32	33	34	35
2	36	37	38	39	40	41	42	43	44	45	46	47
3	48	49	50	51	52	53	54	55	56	57	58	59
4	60	61	62	63	64	65	66	67	68	69	70	71
5	72	73	74	75	76	77	78	79	80	81	82	83
6	84	85	86	87	88	89	90	91	92	93	94	95
7	96	97	98	99	100	101	102	103	104	105	106	107
8	108	109	110	111	112	113	114	115	116	117	118	119
9	120	121	122	123	124	125	126	127	—	—	—	—

cuiti integrati di uso comune (UART) che vengono utilizzati per convertire i dati da parallelo a seriale. Fra le tante possibilità di configurazione del formato dei dati, per il MIDI è stata adottata quella di **figura 3**. Il formato dei dati MIDI prevede un pacchetto di 10 bit costituito da 1 bit di start, 8 bit di dati, 1 bit di stop e nessuna parità.

La velocità è di 31250 Baud (Baud, cioè numero di bit al secondo trasmessi con un flusso di dati continuo); quindi un messaggio di 10 bit viene trasmesso in 320 microsecondi. Mentre gli strumenti musicali effettuano tutte le operazioni di predisposizione a questi standard automaticamente, se si vuole scrivere un programma per un computer collegato a un'interfaccia MIDI, occorre inserire nel software le informazioni necessarie affinché il funzionamento della UART avvenga secondo le specifiche sopra descritte. Ciò viene fatto inviando opportune poke all'indirizzo dell'interfaccia nella cosiddetta fase di inizializzazione, ma ritorneremo più avanti su questo tema. Avendo definito un sistema per l'invio di informazioni di tipo seriale, dobbiamo chiarire ora come collegare fra di loro i vari apparati che costituiscono il sistema e in che modo indirizzare selettivamente i messaggi di informazione. Ci sono due modi fondamentali per effettuare i collegamenti: in serie come in **figura 4** o a stella come in **figura 5**. Il criterio di scelta dipende dal numero di apparati. Con il primo sistema essi si troveranno in serie e quindi i messaggi, essendo ripetuti in ogni strumento, subiranno un ritardo progressivo che alla lunga sarà avvertibile a un orecchio esercitato. Con il sistema a stella occorre un accessorio, chiamato MIDI THRU BOX che consente di inviare le informazioni contemporaneamente a tutti gli strumenti.

Qualunque sia il tipo di collegamento adottato, se mandassimo ora i nostri messaggi di dati dal computer verso gli strumenti, questi ultimi li riceverebbero tutti ed eseguirebbero i comandi.

Per consentire invece di inviare selettivamente solo certi messaggi a determinati strumenti del sistema, è stato istituito un sistema di indirizzi che, codificati nel corpo dei messaggi, costituiscono i canali di informazioni. Nello standard MIDI sono previsti 16 canali numerati da 1 a 16. La **figura 6** illustra i concetti visti sopra con degli esempi. Bisogna fare attenzione nell'acquisto degli strumenti perché mentre il linguaggio MIDI, è previsto per dare una grande flessibilità d'impiego, le possibilità di usarlo in pieno o meno dipendono dal grado di implementazione MIDI dello strumento specifico.

Aldo e Andrea Laus

(continua)

Controlli standard

I numeri a disposizione per il primo byte vanno da 0 a 121. Solo alcuni di questi sono stati standardizzati come da esempio in tabella.

Tipo di controllo	Valore
Ruota o leva di modulazione	1
Controllore a fiato	2
Controllore a pedale	4
Volume principale	7
Pedale sustain	64
Pedale portamento	65

Gestione avanzata dell'alta risoluzione con 41 nuovi comandi superveloci. Hires, Split Screen variabile, scrolling e due pagine distinte completamente indipendenti.

Grafik 2000

La gestione della grafica, in particolare dell'alta risoluzione, è da sempre legata alla programmazione in linguaggio macchina.

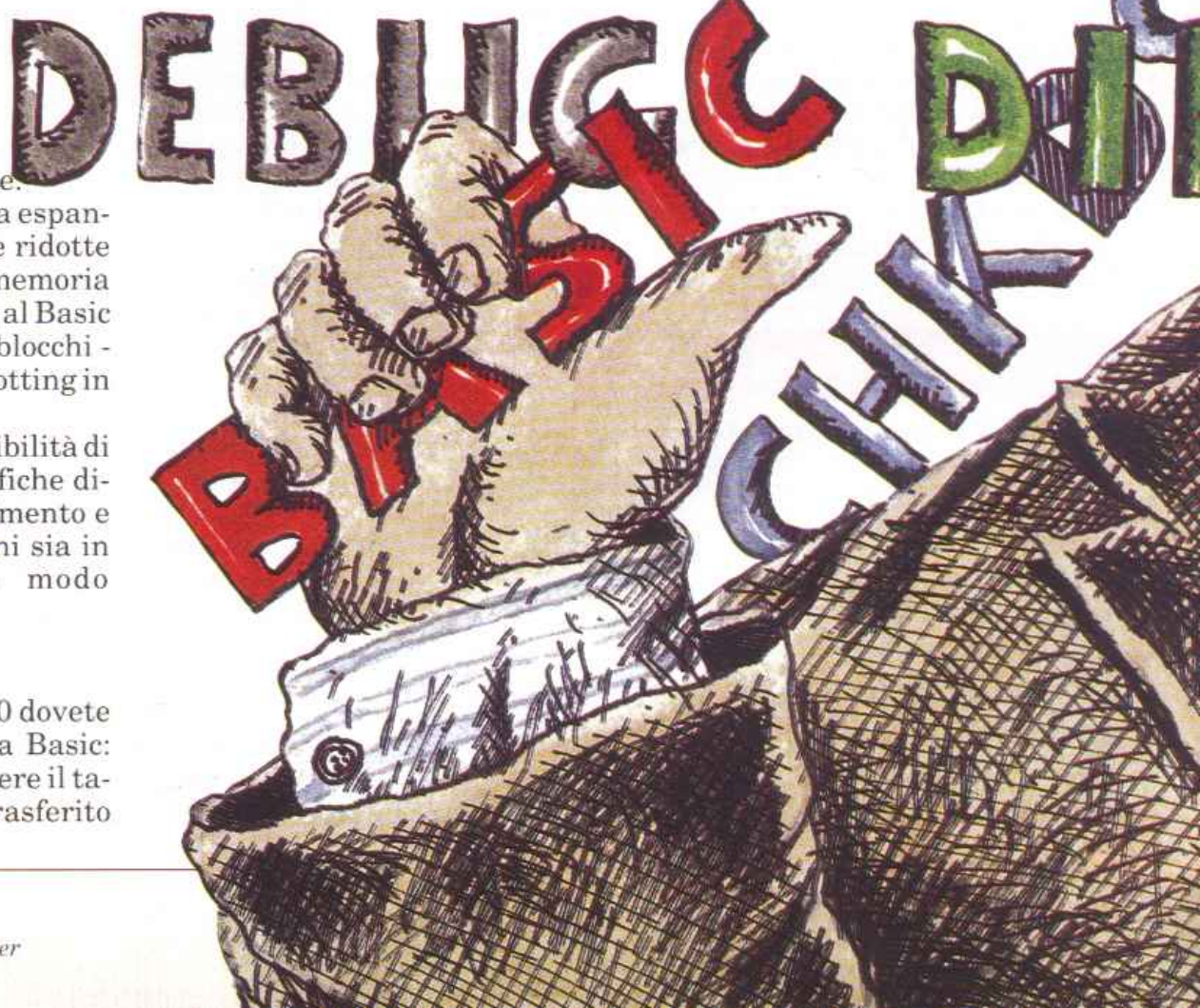
Abbiamo infatti più volte sottolineato come in commercio non esistano vere e proprie espansioni mirate e quindi chi ha bisogno di gestire solo un particolare aspetto del C64 è costretto a costruirsi le routine di cui ha bisogno (e qui spesso si deve ricorrere all'assembler) oppure a servirsi di una mega-espansione, con tutti i problemi che comporta una scelta di questo tipo.

L'espansione di questo mese dovrebbe colmare egregiamente una delle numerose lacune del settore. Grafik 2000 è una completissima espansione per la gestione dell'alta risoluzione.

I punti di forza di questa espansione sono sicuramente le ridotte dimensioni - caricata in memoria non ruba neanche un byte al Basic e su disco occupa solo 27 blocchi - e la notevole velocità di plotting in alta risoluzione.

A ciò si aggiunge la possibilità di utilizzare due pagine grafiche distinte scambiabili a piacimento e di usare tutte le istruzioni sia in modo diretto sia in modo programma.

Come funziona

Per caricare Grafik 2000 dovete digitare la seguente linea Basic: Load"grafik 2000" e premere il tasto Return. Se avete già trasferito l'espansione su disco allora dovete utilizzare l'istruzione: Load"grafik 2000", 8,1. Una volta caricata in memoria, Grafik 2000

si attiva con Run. Ecco l'elenco delle nuove istruzioni aggiunte all'interprete standard del C64:

• **Clear.** Cancella la pagina di alta risoluzione. Non necessita di alcun parametro. Non ha alcun effetto sulla bassa risoluzione e non modifica gli attributi (colore di linea e di fondo) fissati per l'alta risoluzione.

• **Mode.** Permette di passare dal modo testo all'alta risoluzione e viceversa. La sintassi è: Mode n. Con n=0 viene attivato il modo testo mentre con n=1 si entra in alta risoluzione. L'istruzione non ha alcun effetto su ciò che si trova nel modo grafico attivato. Grafik 2000 non mette a disposizione nessuna istruzione per l'impiego dell'alta risoluzione multicolore. Tuttavia non fa pesare questa carenza poiché permette di utilizzare con semplicità ed efficienza più colori in alta risoluzione standard.

• **Color.** Consente di fissare i colori di linea e di fondo per l'alta risoluzione. La sintassi è: Color l, f dove l e f sono rispettivamente i codici dei colori di linea e di fondo che si vogliono impostare.

• **Change.** Permette di scambiare fra loro le due pagine grafiche a disposizione. Può essere utilizzata anche se è stata attivata una finestra di alta risoluzione nel modo testo.

• **Invers.** Inverte il contenuto della pagina in alta risoluzione. Non necessita di alcun parametro.

• **Comb.** Determina la modalità di funzionamento delle istruzioni che agiscono sull'alta risoluzione. La sintassi è: Comb n dove n è un intero che può assumere solo i valori 1, 2 e 3.

Ecco il significato di ciascun valore:

n=1: Or logico.

n=2: And logico.

n=3: Or esclusivo.

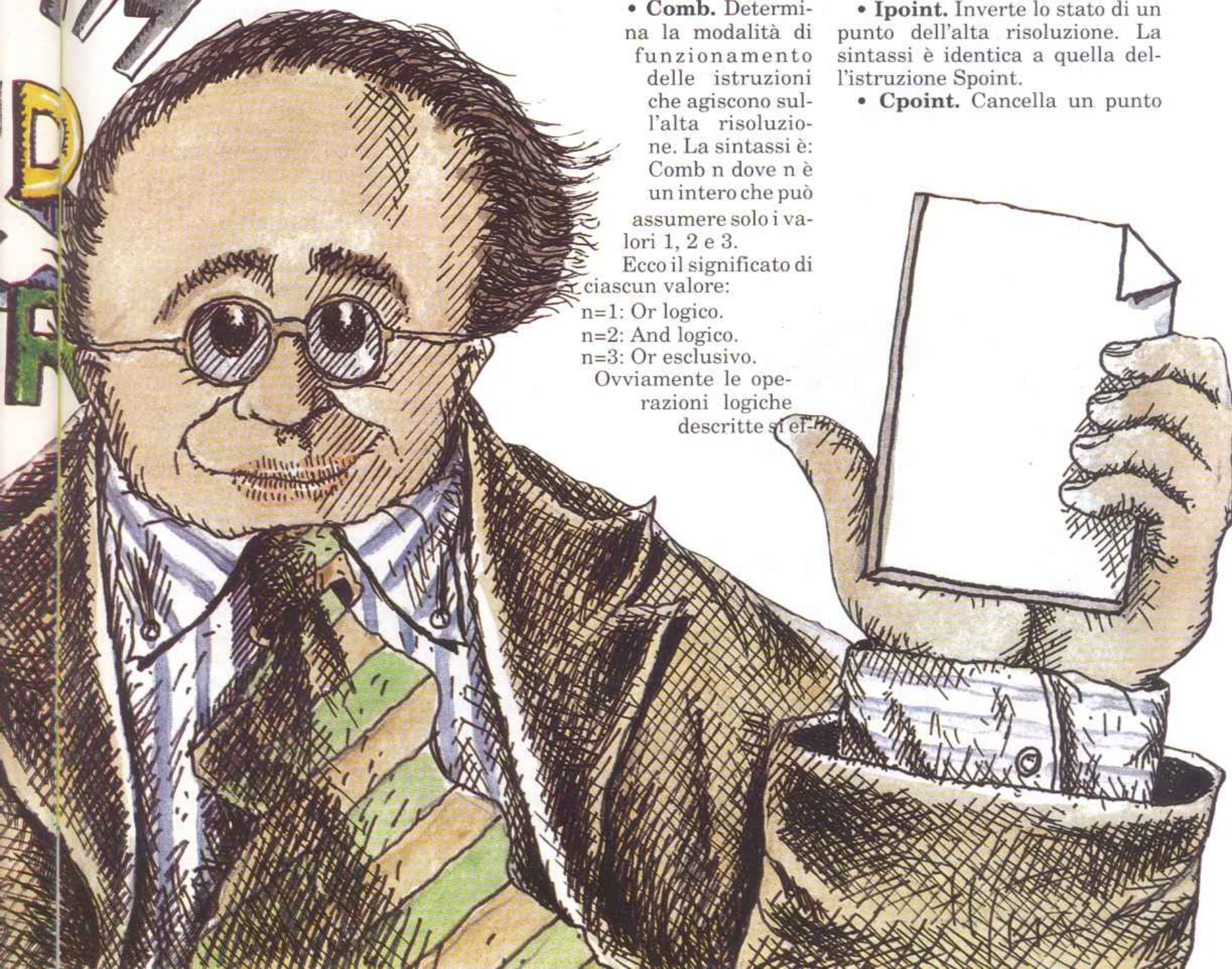
Ovviamente le operazioni logiche descritte si effettuano fra i punti a cui si accede e quelli già presenti in alta risoluzione.

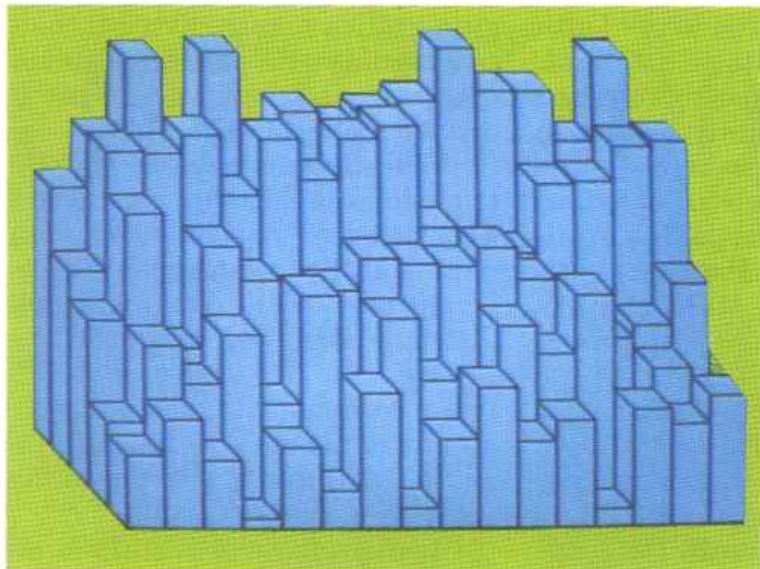
• **Gsave.** Permette di salvare su disco o nastro una schermata di alta risoluzione. La sintassi è: Gsave a$, dvn dove a$ è il nome del file su cui verrà salvata la pagina grafica mentre dvn è il codice associato alla periferica su cui effettuare il salvataggio (8 per il drive e 1 per il registratore).

• **Gload.** È l'istruzione simmetrica della precedente e consente di caricare da nastro o disco una pagina grafica salvata con Gsave. La sintassi è identica a quella dell'istruzione precedente.

• **Spoint.** Setta un punto dell'alta risoluzione. La sintassi è: Spoint x, y dove x e y rappresentano le coordinate del punto da settare.

• **Ipoint.** Inverte lo stato di un punto dell'alta risoluzione. La sintassi è identica a quella dell'istruzione Spoint.

• **Cpoint.** Cancella un punto

dell'alta risoluzione. La sintassi è identica a quella dell'istruzione precedente.

• **Pscline.** Questa istruzione si rivela particolarmente utile per creare effetti 3D. Setta un punto dell'alta risoluzione e automaticamente cancella tutti i punti che si trovano sulla retta parallela ai bordi e passante per il punto stesso.

• **Hmark.** Ha un ruolo simile all'istruzione Spoint. Hmark permette infatti di settare contemporaneamente 5 punti consecutivi in orizzontale. La sintassi è: Hmark x, y dove x e y sono le coordinate del punto medio dell'insieme di punti da settare.

• **Vmark.** È analoga alla precedente con la sola eccezione che agisce in direzione verticale.

• **Hline.** Permette di tracciare una linea orizzontale. La sintassi è: Hline y dove y rappresenta l'ordinata costante di tutti i punti che compongono la linea.

• **Vline.** Permette di tracciare una linea verticale. La sintassi è: Vline x dove x rappresenta l'ascissa comune a tutti i punti della linea.

• **Sline.** Traccia una linea fra due punti. La sintassi è: Sline x0, y0, x1, y1. Ovviamente i quattro parametri rappresentano le coordinate degli estremi del segmento da tracciare.

• **Cline.** Cancella una linea. La sintassi è identica a quella dell'istruzione precedente.

• **Iline.** Inverte lo stato dei punti di una linea. La sintassi è identica a quella dell'istruzione Sline.

• **Scircle.** Permette di tracciare un'ellisse. La sintassi è: Scircle x, y, a, b. I parametri x e y sono le coordinate del centro dell'ellisse mentre i due parametri successivi rappresentano, rispettivamente, l'ampiezza del semiasse orizzontale e verticale. Per ottenere un cerchio basta porre a=b.

• **Ccircle.** Cancella tutti i punti su una circonferenza. La sintassi è identica a quella dell'istruzione precedente.

• **Icircle.** Inverte tutti i punti su una circonferenza. La sintassi è identica a quella dell'istruzione Scircle.

• **Sellipse.** È una delle istruzioni più complesse dell'intera espansione. Permette di tracciare archi di ellisse di ampiezza variabile. La sintassi è: Sellipse x, y, a, b, k, st, en, inc. I primi quattro parametri hanno un significato analogo ai parametri omonimi dell'istruzione Scircle.

Il parametro k rappresenta il punto di riferimento per il tracciamento del segmento di ellisse. I due parametri successivi sono il punto di partenza e d'arrivo, in radianti, dell'arco di ellisse da tracciare. L'ultimo parametro rappresenta l'incremento, in radianti, fra i punti del segmento di ellisse.

• **Cellipse.** Cancella tutti i punti su un segmento di ellisse. La sintassi è identica a quella dell'istruzione precedente.

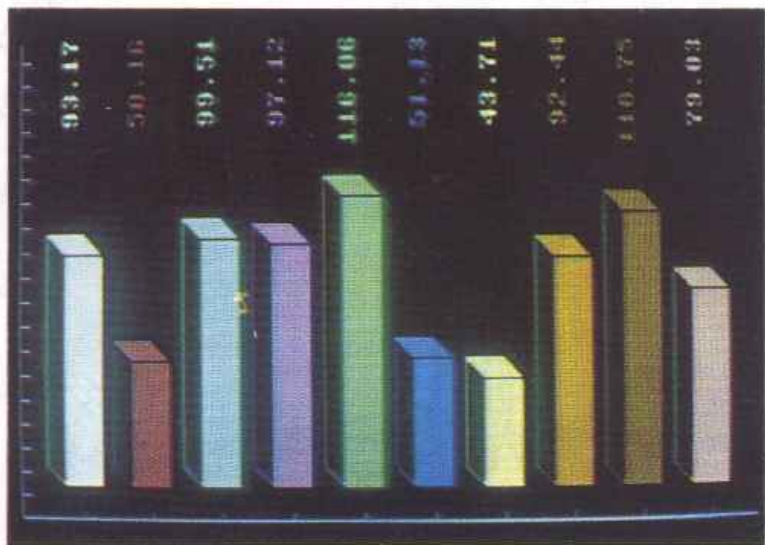
• **Iellipse.** Inverte tutti i punti su un segmento di ellisse. La sintassi è identica a quella dell'istruzione Sellipse.

• **Stext.** Permette di visualizzare una stringa in alta risoluzione. La sintassi è: Stext vr, ln, al, or, sp, x, y, a$. Il primo parametro, dr indica il verso di scrittura, cioè la modalità di visualizzazione della serie di caratteri. Questo parametro può assumere solo i seguenti valori:

vr=1: da sinistra a destra.
vr=2: da destra a sinistra.
vr=3: dal basso verso l'alto.
vr=4: dall'alto verso il basso.

Il parametro ln indica la larghezza dei caratteri e può assumere un valore qualsiasi fra 1 (larghezza normale) e 25 (larghezza massima). Il parametro al indica invece l'altezza dei caratteri e può variare anch'esso fra 1 (altezza normale) e 25 (altezza massima).

Il parametro or indica invece l'orientamento della stringa cioè l'angolo, espresso in radianti, che la linea immaginaria su cui poggiano i caratteri della stringa forma con la verticale. Sp rappresenta la spaziatura fra i caratteri e può variare fra 0 e 199 (per una spaziatura normale sp=8).

I due parametri successivi rappresentano le coordinate del primo carattere della stringa, o meglio le coordinate da cui partirà la visualizzazione della stringa

mentre a$ è l'insieme di caratteri da visualizzare.

• **Ctext.** Cancella una serie di caratteri. Ha la stessa sintassi dell'istruzione precedente.

• **Itext.** Inverte una serie di caratteri. Ha la stessa sintassi dell'istruzione Stext.

• **Fill.** Riempie un'area delimitata da una linea chiusa. La sintassi è: Fill x, y dove x e y rappresentano le coordinate di un punto qualsiasi all'interno della superficie chiusa.

• **Duplicate.** Permette di duplicare una parte di alta risoluzione. La sintassi è: Duplicate x0, y0, x1, y1, x, y. I primi quattro parametri definiscono il blocco di alta risoluzione che si vuole copiare e rappresentano rispettivamente le coordinate dell'angolo superiore sinistro e di quello inferiore destro. Gli ultimi due parametri invece sono le coordinate del punto da cui inizierà la trascrittura del blocco selezionato.

• **Scroll.** Effettua lo scrolling di una parte di alta risoluzione. La sintassi è: Scroll r, x0, y0, x1, y1. Il primo parametro indica la direzione dello scrolling. Ecco i valori che il parametro può assumere:

r=1: scrolling a destra.
r=2: scrolling verso sinistra.
r=3: scrolling verso il l'alto.
r=4: scrolling verso il basso.

Gli ultimi quattro parametri definiscono la porzione di alta risoluzione da scrollare. Tenete presente che lo scrolling avviene sempre con wrap round.

• **Window.** Permette di aprire una finestra in alta risoluzione sullo schermo in modo testo. La sintassi è: Window y0, y1 dove y0 e y1 sono la prima e l'ultima linea della porzione di schermo che si vogliono porre in alta risoluzione. Ovviamente i due parametri possono assumere solo valori compresi fra 0 e 199. Tenete presente che quando è stata attivata una finestra di alta risoluzione non è più possibile utilizzare l'istruzione Mode per attivare o disattivare l'alta risoluzione. Per disattivare una finestra dovete usare l'istruzione Window senza parametri.

• **Lowcol.** Permette di definire i colori di linea e di fondo di un insieme di 8x8 punti nella pagina in alta risoluzione. La sintassi è: Lowcol x, y, l, f.

I primi due parametri rappresentano le coordinate di un punto qualsiasi dell'insieme di 8x8 punti mentre l e f sono i codici dei colori di linea e di fondo da associare alla porzione di alta risoluzione selezionata.

• **Sprite.** Consente di posizionare uno sprite sullo schermo. La sintassi è: Sprite n, x, y dove n rappresenta il numero dello sprite che si vuole porre sul video mentre x e y rappresentano le coordinate dell'angolo superiore sinistro dello sprite.

• **Ssave.** Permette di salvare su nastro o disco la definizione di uno sprite. La sintassi è: Ssave n, n$, dvn. Il primo parametro rappresenta il numero associato allo sprite da salvare mentre i due parametri successivi sono il nome del file su cui verrà salvata la definizione dello sprite e il codice del supporto su cui effettuare il salvataggio.

• **Sload.** Permette di caricare la definizione di uno sprite. La sintassi è identica a quella dell'istruzione precedente.

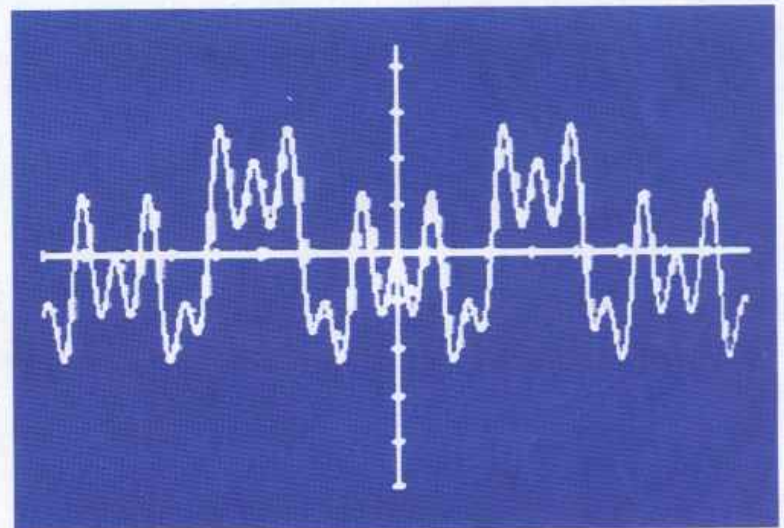
• **Pscreen.** Permette di attivare la pagina grafica su cui avranno effetto le istruzioni dell'espansione. La sintassi è: Pscreen n dove n può assumere i valori 1 e 2. Tenete presente che sulla pagina 2 non hanno effetto le istruzioni: Sprite, Scroll, Duplicate, Test e Ctest. Per scambiare le due pagine grafiche a disposizione dovete usare l'istruzione Change.

• **Tscreen.** Questa istruzione come la precedente permette di selezionare una delle due pagine grafiche a disposizione. Attivando la pagina 2 con questa istruzione si hanno a disposizione solo le istruzioni non disponibili in questa pagina viste con l'istruzione precedente.

• **Colplot.** Setta il colore di visualizzazione e di fondo per i punti a cui si accede con le istruzioni: Spoint, Sline, Scircle, Sellipse, Stext, Fill, Scroll, Duplicate. La sintassi è: Colplot l, f dove l e f sono i codici del colore di linea e di fondo scelti.

• **Plot.** Permette di attivare l'effetto 3D. La sintassi è Plot n dove n può assumere solo i seguenti valori:

n=1: effetto 3D disattivato.
n=2: effetto 3D attivato.

Tenete presente che l'effetto 3D ha effetto solo con le seguenti istruzioni: Spoint, Sline, Scircle, Sellipse, Stext, Scroll, Duplicate.

• **Test.** È una funzione che restituisce lo stato di un punto della pagina grafica. La sintassi è: Test(x,y) dove x e y sono le coordinate del punto che si vuole testare. Se il punto è settato la funzione restituisce il valore 1, in caso contrario il valore 0.

• **Ctest.** È una funzione che permette di conoscere il codice del colore di un punto della pagina grafica. eLa sintassi è identica a quella dell'istruzione precedente e ovviamente il valore restituito corrisponde al codice del colore del punto selezionato.

Daniele Maggio

Hardware e funzionamento del C64

Diamo un'occhiata all'interno del Commodore 64. Vedremo i principali integrati, i segnali fondamentali e discuteremo i pregi e i difetti dell'architettura dell'home computer più diffuso nel mondo.

Ingrandimento delle connessioni del microprocessore.

In questa serie di articoli esamineremo la struttura hardware del C64, la collocazione dei vari integrati, la loro pinatura e la funzione che svolgono nel calcolatore. Si darà per scontata una minima conoscenza di base del funzionamento dei sistemi a microprocessore e della loro programmazione.

Struttura del C64

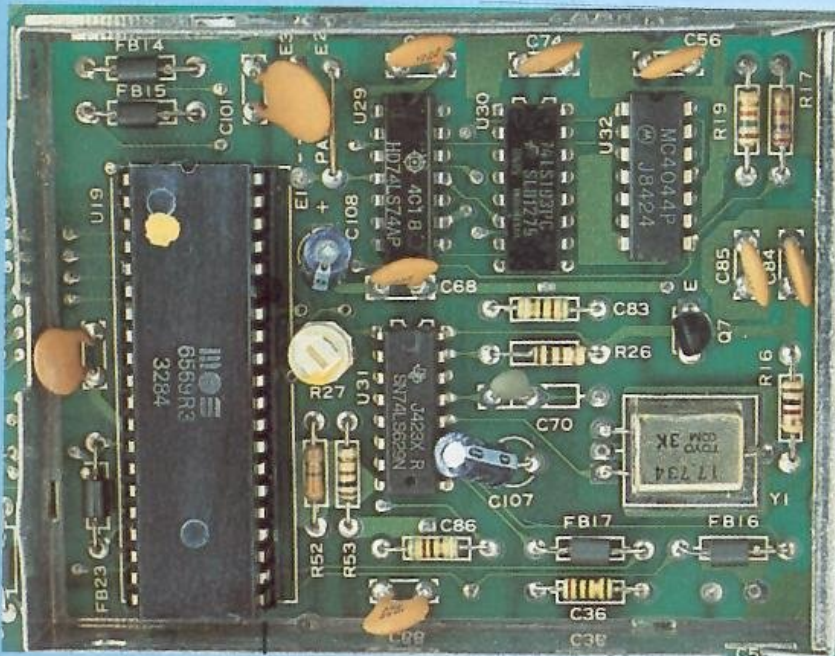
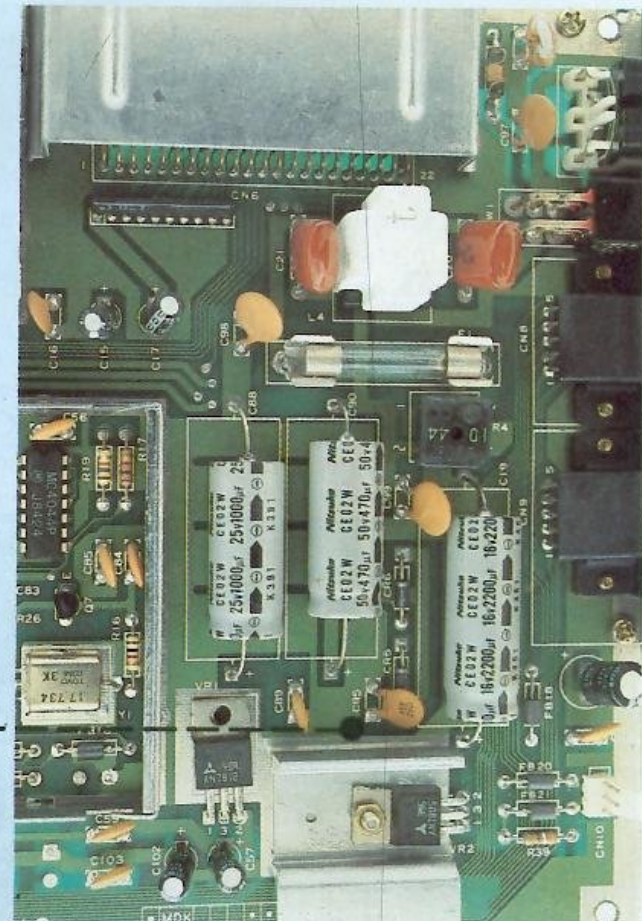
Si è scelto come esempio uno dei modelli meno recenti ma più venduti. Eventuali modifiche, apportate ai modelli successivi, verranno illustrati quando se ne presenterà l'occasione. Nella **figura 1** sono rappresentate la piastra madre del C64 e l'ubicazione dei principali integrati. La circuiteria del C64 può essere divisa in blocchi correlati e interdipendenti:

1) Circuiti di alimentazione;
2) Generatore del segnale di Clock;

Veduta d'insieme dello chassis del C64.

3) Circuito di reset;
4) Microprocessore;
5) PAL e decodifica gli indirizzi;
6) ROM
7) RAM e multiplexer;
8) Interfacce
9) Logica di controllo Restore;
10) Controllo del registratore;
11) Controllo del bus seriale;
12) Modulatore UHF.

Il cuore del C64 è il microprocessore, il cui compito consiste nell'eseguire i programmi che non sono altro che sequenze di istruzioni memorizzate nelle memorie RAM e ROM. Durante l'esecuzione il microprocessore elaborerà dei dati presenti nelle memorie o li invierà verso le interfacce (output) o li riceverà da queste (input). Nei computer sono presenti sia memorie RAM che memorie ROM. Le prime sono memorie ad accesso casuale; le seconde sono memorie di sola lettura.

Nel C64 sono presenti tre ROM: la ROM Basic che contiene le routine dell'interprete Basic; la ROM Kernal che contiene il codice delle procedure che gestiscono le operazioni di I/O; la ROM Caratteri che contiene le informazioni necessarie a visualizzare ogni carattere sullo schermo. Le RAM contengono temporaneamente i dati prima e dopo la loro elaborazione e occupano nel C64 ben 64 Kb, anche se normalmente, a causa della presenza nella mappa di memoria delle ROM e interfacce, se ne usano solo 48 Kb. Nel calcolatore sono anche presenti le interfacce il cui scopo è quello di comunicare con l'esterno. Nel C64 ve ne sono quattro: l'interfaccia video detta anche VIC il cui compito principale è quello di generare il segnale video necessario alla visualizzazione di quanto appare sullo schermo. Il SID, che si occupa della sintesi di tutti i suoni, nonché dell'elaborazione del segnale proveniente dalle paddles. CIA1 si occupa della

Ingrandimento del generatore di clock.

Ingrandimento dell'alimentatore interno.

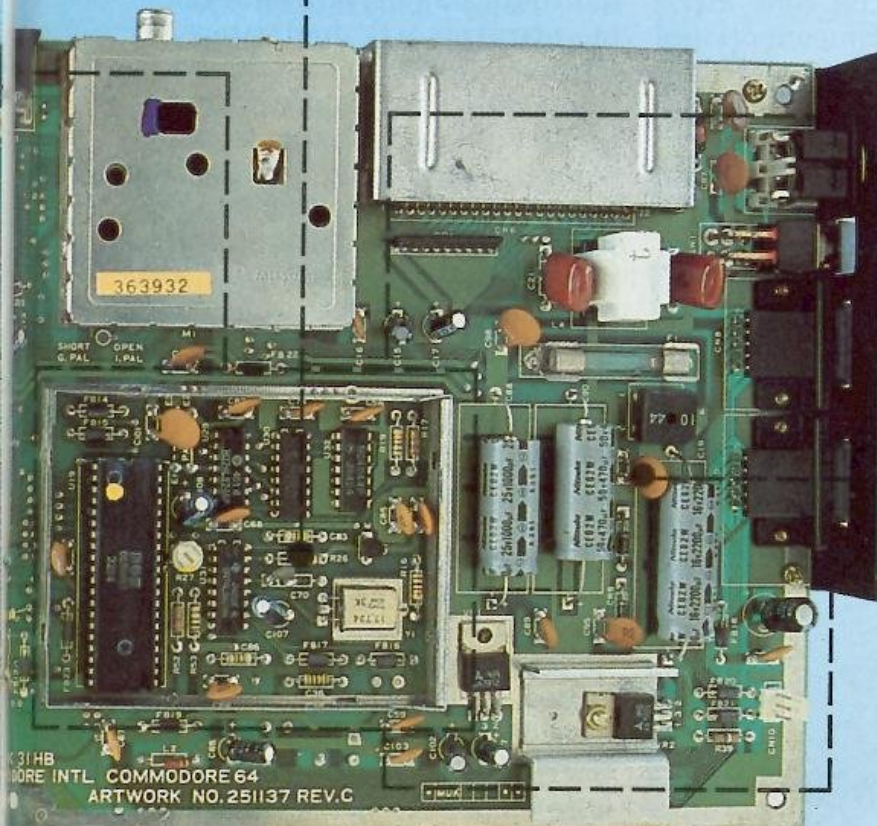
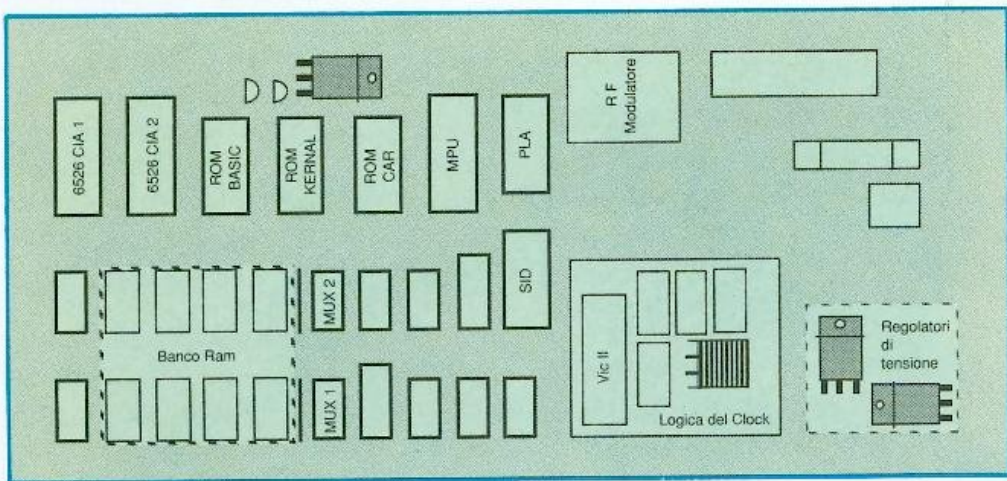



Figura 1. Veduta d'insieme dello chassis del C64.

scansione della tastiera, ovvero di rivelare quali tasti vengano premuti, mentre CIA2 gestisce la porta parallela e quella seriale, garantendo le comunicazioni con dispositivi esterni quali disk driver, stampanti e modem.

Alimentazione

I circuiti di alimentazione sono dislocati sia nell'alimentatore esterno sia sulla piastra madre. L'alimentatore esterno fornisce al C64 due tensioni: 9 volt in alternata con un assorbimento massimo di 1 ampère e 5 volt in tensione continua e stabilizzata con un assorbimento massimo di 1,5 ampère. Lo schema elettrico di tale alimentatore esterno è rappresentato nella **figura 2.** Come si può vedere la tensione di rete (220 volt) attraverso il fusibile F1 arriva al primario di un trasformatore. Sui due secondari di detto trasformatore saranno presenti due tensioni alternate di circa 9 volt. Una di queste verrà inviata direttamente alla piastra madre del C64, mentre l'altra verrà raddrizzata dai 4 diodi (D1, D2, D3 e D4) e livellata dal condensatore C1 da 4700 microfarad. Successivamente la tensione continua prelevata ai capi di questo condensatore verrà inviata al regolatore di tensione IC1 (un ua7805) il quale fornirà una tensione stabilizzata a 5 volt. La **figura 3** rappresenta l'alimentatore presente sulla piastra madre del C64, nonché lo schema elettrico. Le sigle attribuite ai diversi componenti potrebbero anche non collimare, nel caso il vostro C64 fosse di un modello diverso dal nostro. All'interno del C64, attraverso l'apposito connettore arriveranno le due tensioni generate dall'alimentatore esterno. Di queste, quella stabilizzata a 5 volt viene utililizzata, dopo essere passata attraverso il filtro passa basso costituito da C97, L5 e C100 e l'interruttore di alimentazione SW1, per alimentare la quasi totalità dei circuiti logici. La tensione alternata a 9 volt verrà fatta passare attraverso il filtro antidisturbi costituito da C20, L4, C21, C98 e C99, indi attraverso un secondo scambio dell'interruttore di alimentazione SW1. L'altro ramo di detta tensione alternata passerà invece attraverso il fusibile F1 da 1 ampère. Sucessivamente dalla tensione alternata, attraverso il ponte raddrizzatore CR4, i condensatori C19 e C95, il regolatore di tensione VR2 (un altro ua7805), e i condensatori C102 e C103 verrà ottenuta una tensione continua, stabilizzata al valore di 5 volt, necessari per alimentare i circuiti di clock del C64. La tensione alternata di 9 volt, sarà anche raddrizzata dal diodo CR6 per ottenere una tensione continua, non stabilizzata, a 9 volts, necessaria all'alimentazione della circuiteria di controllo del motore del registratore. Sempre da tale tensione alternata a 9 volt, sarà sintetizzata, attraverso il duplicatore di tensione costituito da C90, CR5, C88, C89, e il regolatore di tensione VR1 (un ua7812) una tensione stabilizzata al valore di 12 volt, necessaria al funzionamento della parte analogica delle interfacce video e audio (VIC e SID). I condensatori C57 e C59 serviranno a livellare ulteriormente questa tensione. Il diodo led che figura sul pannello superiore del C64 è collegato tramite la resistenza R39 all'alimentazione a +5 volt.

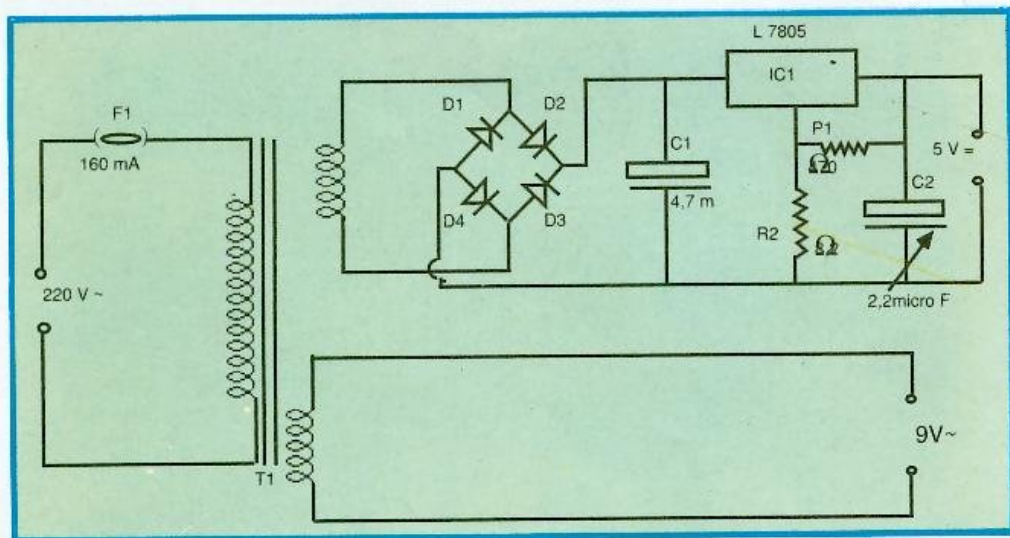



Figura 2. Schema elettrico dell'alimentatore esterno.

Generatore di clock

In un calcolatore il segnale di clock è un segnale di tempificazione il cui scopo è quello di dare il tempo affinché tutte le diverse funzioni, svolte dai vari integrati, siano sincronizzate. Nel C64 di questi segnali di clock ve ne sono ben quattro:

• Color Clock: sincronizza l'informazione colore per ogni singolo pixel all'interno dell'interfaccia video;
• Dot Clock: sincronizza la visualizzazione dei singoli pixel;
• Clock0: segnale in ingresso al microprocessore che ne tempifica tutte le operazioni interne;
• Clock1: segnale in uscita al microprocessore che sincronizza le varie interfacce: SID, CIA1 e CIA2.

Il generatore del segnale di clock, nonché lo schema elettrico che lo rappresenta sono visibili nella **figura 4**. La circuiteria del clock nel C64 è inserita, congiuntamente all'interfaccia video, in una scatoletta metallica, che svolge funzione di schermo elettromagnetico, ed ha una alimentazione autonoma a 5 volt stabilizzati, al fine di garantire la massima stabilità dei segnali generati. Essa si compone di quattro integrati: il primo, U31 (74LS629), è un oscillatore controllato da un quarzo da 14.318.180 Mhz per il sistema americano NTSC o 17.734.472 nel sistema europeo PAL. La frequenza così generata è detta Color Clock e verrà utilizzata dal VIC (interfaccia video), per generare il segnale della cromia (Colore). Gli integrati U29 e U30 costituiscono nell'insieme un contatore binario programmabile, il cui modulo potrà essere variato agendo su un apposito ponticello da modificare a seconda che si intenda usare il sistema NTSC o PAL. Dopo essere stato diviso da U30, e successivamente dal flip flop U29 il segnale, ripassando nuovamente attraverso la seconda metà dell'integrato U31 raggiungerà il VIC. Questo segnale è detto Dot Clock e servirà per ottenere il segnale della luminosità nel VIC. Nelle versioni più recenti del C64 tutta la logica necessaria alla sintesi di questi due segnali di clock è stata integrata in un unico circuito: un MOS 8701. Successivamente all'interno dell'interfaccia video il segnale Dot clock verrà ulteriormente diviso per otto, onde ottenere il segnale Clock0, ovvero il principale segnale di clock del sistema che, alla frequenza di 1.02 Mhz (PAL) e 0,98 Mhz (NTSC), verrà inviato al microprocessore. Il microprocessore produrrà in uscita un segnale sempre alla stessa frequenza di clock0 ma con forma d'onda diversa: clock1, necessario alla sincronizzazione delle diverse interfacce ad esso collegate. Oltre a Clock0 il VIC provvede anche alla sintesi di diversi segnali di controllo il cui compito è quello di amministrare il flusso dei dati e degli indirizzi sui vari bus. Questi segnali sono: AEC, CAS e RAS.CAS e RAS e servono essenzialmente per pilotare i multiplexer e le RAM, e verranno analizzati nei dettagli più avanti, mentre AEC ha il compito di commutare l'uso dei bus fra il microprocessore e l'interfaccia video.

Figura 3. Schema elettrico dell'alimentatore interno.

Circuito di Reset

All'accensione del C64 il contenuto delle RAM è del tutto casuale e così pure lo stato dei registri interni al microprocessore e alle varie interfacce. Per poter inizializzare tutti questi dispositivi si rende necessario abbassare un apposito segnale detto Reset. Quando questo segnale tornerà alto, il microprocessore inizierà l'esecuzione del programma d'inizializzazione. La circuiteria del Reset provvede a generare automaticamente questo segnale all'accensione del calcolatore. Essa è costituita da due integrati e relativi componenti passivi annessi. Il primo integrato è un NE556 (che praticamente è l'unione di due NE555 in un unico chip) e svolge funzione di monostabile, infatti, all'atto dell'accensione del calcolatore il condensatore C24 comincerà a caricarsi tramite la resistenza R34 e, raggiunto un valore di tensione di circa 1,6 volt farà scattare un comparatore, presente all'interno dello stesso integrato che abbasserà l'uscita situata al piedino n.9. Il segnale così generato verrà successivamente in-

viato a una delle sei porte invertenti presenti nell'integrato U8 (SN7406), per essere invertito, indi sarà inviato al microprocessore e alle interfacce SID, CIA1 e CIA2 al fine di resettare tutti questi dispositivi. Osservando questo segnale con un'oscilloscopio si potrà notare come si presenti basso all'istante dell'accensione per poi alzarsi dopo qualche centinaio di millisecondi. Il segnale di Reset oltre ad essere inviato ai vari chip interni al C64 verrà anche propagato sul bus seriale al fine di resettare eventuali dispositivi esterni: disk drive, stampante ecc. Questo tipo di collegamento presenta lo svantaggio che un eventuale segnale spurio di Reset, proveniente dall'esterno, potrebbe resettare il calcolatore. Per sopperire a questo problema nelle versioni più recenti del C64 si è provveduto a differenziare le linee di Reset interno ed esterno, aggiungendo un transistor, svolgente funzioni di invertitore logico.

Il microprocessore

Un microprocessore legge delle istruzioni dalla memoria del calcolatore in cui opera e successivamente le traduce in operazioni elementari che esegue sui dati presenti in memoria o nelle interfacce. Per leggere i codici delle istruzioni e i dati, l'MPU (microprocessore), utilizza un insieme di linee elettriche (detto bus dati), attraverso le quali l'informazione, sotto forma di impulsi elettrici, viene trasferita all'interno del microprocessore. Lo stesso bus dati verrà anche usato per operazioni di scrittura, nel qual caso le informazioni saranno trasferite dai registri interni della MPU a quelli di qualche dispositivo esterno. Per quest'ultima ragione il bus dati è bidirezionale, ovvero percorribile dai segnali in entrambi i sensi. Una linea elettrica, detta RW (lettura o scrittura), in uscita dal microprocessore, indicherà il senso di trasferimento delle informazioni lungo tale bus. Poiché in un sistema a microprocessore esistono più locazioni di memoria, l'MPU, per accedere ad una locazione specifica, sia in lettura che in scrittura, dovrà comunicare al sistema l'indirizzo di tale locazione, tramite dei segnali su di un apposito bus (bus indirizzi). Poiché gli indirizzi, sotto forma di impulsi elettrici, si propagheranno sempre dal microprocessore verso i dispositivi esterni, il bus indirizzi è detto unidirezionale. La **figura 6** rappresenta il microprocessore e lo schema elettrico delle connessioni dei vari pin.

La CPU del C64

Il microprocessore che governa il C64 è un MOS 6510; questo circuito può essere immaginato co-

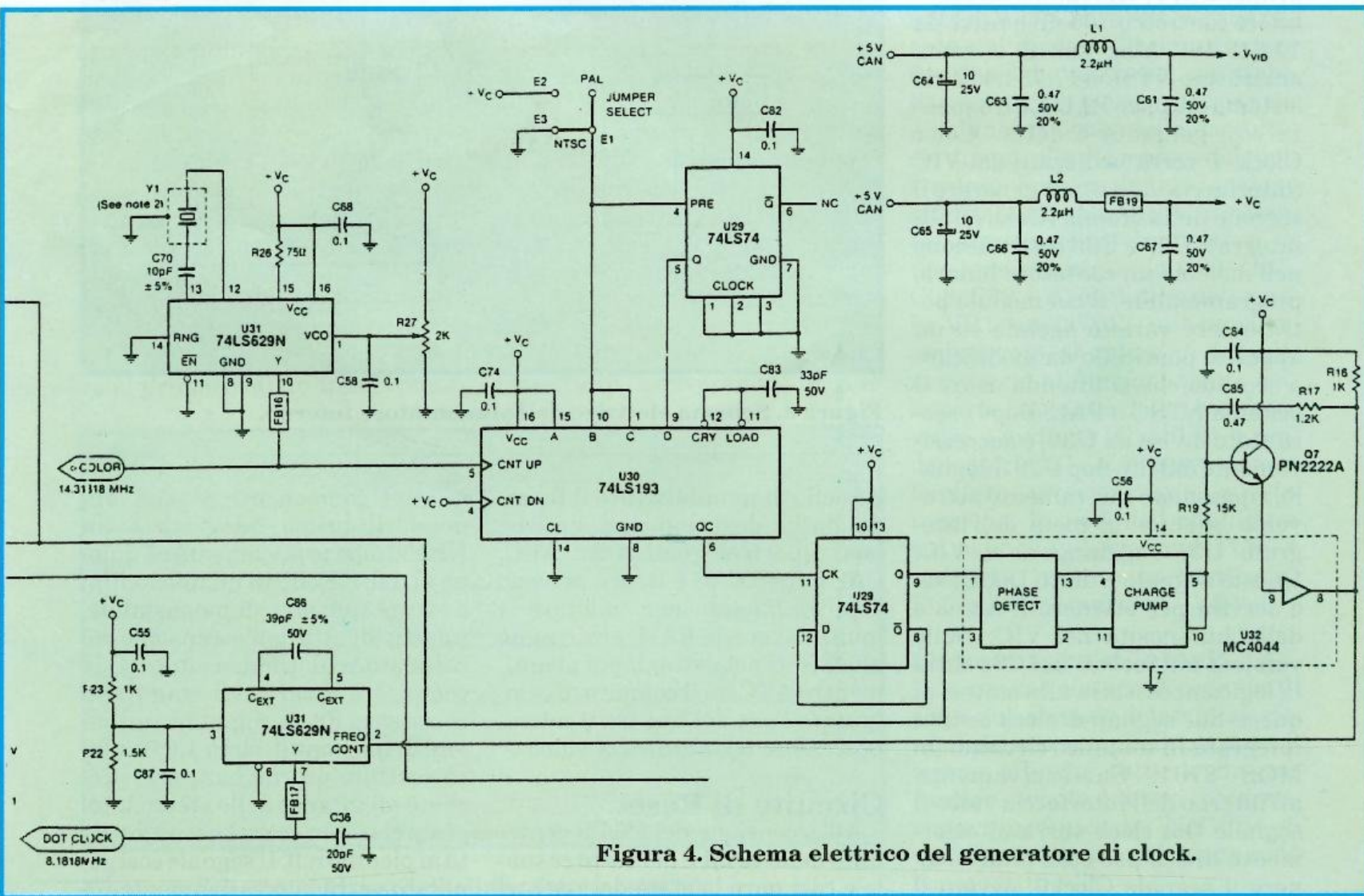



Figura 4. Schema elettrico del generatore di clock.

me l'unione di un microprocessore 6502 e un interfaccia parallela a 6 bit, raggruppati in un unico chip. Il microprocessore per poter funzionare necessita di un segnale di temporizzazione: clock 0 e ne genera a sua volta un altro: clock 1, cheviene inviato alle interfacce e al PLA. Altro segnale in entrata al microprocessore è quello di Reset, che ne inizializza il funzionamento. Il bus dati del microprocessore, costituito da otto linee elettriche in grado di supportare un'informazione nel formato byte, è collegato internamente all'interfaccia parallela ed esternamente a tutte le memorie ROM, a tutte le interfacce e alla RAM. Il microprocessore, per poter accedere ai vari registri, siano essi ubicati nella RAM, nelle ROM o nelle interfacce, deve fornire su un apposito bus l'indirizzo della locazione che intende leggere o scrivere. Questo bus, detto bus indirizzi, è costituito da 16 linee elettriche, che potranno portare un'informazione complessiva di 16 bit, ovvero potranno indirizzare 65536 registri diversi che saranno numerati, in esadecimale, da $0000 a $FFFF. Il bus indirizzi verrà inviato ai multiplexer, alle ROM e alle interfacce nella sua parte meno significativa, mentre i bit di peso maggiore saranno inviati al PLA che provvederà alla decodifica dell'indirizzo. Anche il 6510 è dotato di una linea di selezione lettura/scrittura: RW, che verrà inviata al PLA, per raggiungere successivamente le RAM; e di varie interfacce: VIC, SID, CIA1 e CIA2. Le ROM, essendo dispositivi di sola lettura, non necessitano di questo segnale.

Condivisione del bus

Le RAM all'interno del C64 dovranno essere accessibili non solo al microprocessore, ma anche all'interfaccia video (VIC), infatti quest'ultimo dispositivo dovrà leggervi i dati relativi sia ai caratteri da rappresentare sullo schermo sia agli sprite. Poiché il C64 è dotato di un unico bus dati e un unico bus indirizzi, questi dovranno essere utilizzati alternativamente dalla MPU e dal VIC. Per l'esattezza ognuno dei 2 dispositivi utilizzerà i bus per mezzo ciclo di clock (clock0), pari a circa 500 ns, lasciando i bus liberi per l'altro dispositivo il rimanente mezzo ciclo di clock.

Il segnale che commuta l'uso del bus fra il microprocessore e l'interfaccia video è AEC, viene generato dallo stesso VIC e successivamente inviato alla MPU attraverso una porta AND. Quando il segnale AEC sarà alto, il bus indirizzi, il bus dati e la linea RW del microprocessore saranno collegati ai rispettivi bus del calcolatore, mentre quando AEC sarà basso, i bus e RW della MPU risulteranno essere nello stato ad alta impedenza, ovvero scollegati dai bus di sistema, permettendo a questi ultimi di essere usati, per operazioni di lettura, dall'interfaccia video. Malgrado abbia a disposizione metà del ciclo di clock0 per accedere ai dati, durante la visualizzazione degli sprite, e la lettura della memoria di schermo, il VIC si vede costretto, dovendo accedere alla memoria anche durante il tempo in cui il segnale di clock0 è alto, a bloccare temporaneamente il microprocessore. Operazione che viene compiuta abbassando il segnale BA, in uscita dal VIC, che tramite una porta AND abbasserà il segnale RDY, determinando l'arresto momentaneo del microprocessore.

In contemporanea il VIC abbasserà anche AEC, allo scopo di poter disporre dell'uso dei bus, isolando il microprocessore.

D.M.A.

Come molti calcolatori più grandi anche il C64 consente l'esecuzione di operazioni di DMA, ovvero è possibile per un dispositivo esterno accedere direttamente ai vari registri, siano essi in memoria o nelle interfacce. Tale accesso sarà consentito soltando nei semicicli alti di AEC, ovvero nei tempi comunemente utilizzati dal microprocessore per accedere ai dati tramite i bus. Il dispositivo esterno, prima di accedere ai bus interni, dovrà abbassare il segnale DMA, presente nella porta espansioni; tale segnale, passando attraverso due porte AND, determinerà l'abbassamento sia di AEC che di RDY sul microprocessore, bloccandolo e isolandolo dai bus, affinché non interferisca con le operazioni di DMA.

Figura 5. Schema elettrico della logica di Reset.

Gli interrupt

Il microprocessore MOS 6510, come del resto il 6502, possiede un segnale di interruzione: IRQ che, abilitato nello stato basso, determina l'esecuzione di una procedura d'interruzione. Ovvero la MPU terminerà l'istruzione in corso, salverà il contatore di programma sullo stack, e intraprenderà l'esecuzione di una routine il cui indirizzo sarà prelevato alle locazioni $FFFE $FFFF.

Il sistema delle interruzioni è usato nel C64 per diversi scopi: il lampeggio del cursore, il controllo della tastiera, il controllo e l'av-

viamento del registratore nonché le operazioni di lettura e scrittura su cassette magnetiche. Tutte queste operazioni verranno esaminate più avanti, quando ci si occuperà in dettaglio del funzionamento delle diverse interfacce. Per ora ci basti sapere che il segnale IRQ è visto come entrata dal microprocessore ed è collegato alle uscite IRQ delle interfacce VIC e CIA1.

abbassamento dell'omonimo segnale NMI, presente sul microprocessore. Il segnale NMI potrà essere originato dall'interfaccia CIA2 o dalla logica di controllo del tasto Restore.

Interfaccia a 6 bit

Come abbiamo già avuto modo di notare all'interno dello stesso chip del microprocessore è anche presente un'interfaccia parallela

Figura 6. Schema delle connessioni del microprocessore.

Un altro tipo di interruzione implementata sul microprocessore è quella non mascherabile: NMI che, ad una priorità superiore a IRQ, determina l'interruzione della procedura in corso, il salvataggio sullo stack dell'indirizzo di ritorno e l'esecuzione di una routine il cui indirizzo è contenuto alle locazioni: $FFFA $FFFB. Tale interruzione verrà originata da un

a 6 bit. Questa interfaccia è controllata da due registri: uno che presiede allo stato delle singole linee facenti capo all'interfaccia (detto registro dati) e uno che sovraintende alla direzione dei segnali, ovvero a quali linee siano da considerarsi ingressi e quali uscite: (registro direzione dati). Questi due registri potranno essere scritti e letti direttamente dall'MPU alle locazioni: $0000 e $0001. Le sei linee dell'interfaccia sono usate a diversi scopi: le prime tre controllano la configurazione della memoria, mentre le tre linee di peso maggiore sono utilizzate per il controllo del registratore.

Per l'esattezza la prima linea facente capo a P0, chiamata LoRam e impostata come uscita, permette la disabilitazione della ROM del Basic. Se tale linea è alta nell'area di memoria presente tra $A000 e $BFFF sarà allocata la ROM del Basic, mentre se tale linea risulterà essere bassa, l'area sarà occupata da RAM. La seconda linea, facente capo a P1, di nome HiRam, ed impostata come uscita controllerà invece la ROM del Kernal: se sarà impostata alta nell'area di memoria compresa tra $E000 e $FFFF sarà allocata la ROM del Kernal, viceversa tale area conterrà della RAM. La terza linea di nome Charen, collegata a P2 e programmata come uscita, controllerà invece la ROM dei caratteri. Se tale linea sarà impostata alta, come è normalmente, nell'area tra $D000 e $DFFF saranno indirizzabili i vari chip delle interfacce, mentre se risulterà essere bassa, nella stessa area il microprocessore potrà accedere alla ROM caratteri.

Per quanto riguarda invece il controllo del registratore la linea facente capo a P3, di nome CassWRT, programmata come uscita, controlla la scrittura su nastro, ovvero con il suo alzarsi e abbassarsi origina il segnale che verrà memorizzato sul nastro magnetico dal registratore. La linea collegata a P4, detta CassSense e impostata come ingresso, controlla l'eventuale pressione dei tasti del registratore: a tasto premuto corrisponderà linea abbassata. Infine l'ultima linea, CassMotor, collegata a P5 e programmata come uscita, è collegata alla logica di controllo del motorino del registratore: linea bassa motorino acceso, linea alta motorino spento.

Sergio Fiorentini

(continua)